

JPRS 82389

3 December 1982

USSR Report

CYBERNETICS, COMPUTERS AND
AUTOMATION TECHNOLOGY

No. 65



FOREIGN BROADCAST INFORMATION SERVICE

NOTE

JPRS publications contain information primarily from foreign newspapers, periodicals and books, but also from news agency transmissions and broadcasts. Materials from foreign-language sources are translated; those from English-language sources are transcribed or reprinted, with the original phrasing and other characteristics retained.

Headlines, editorial reports, and material enclosed in brackets [] are supplied by JPRS. Processing indicators such as [Text] or [Excerpt] in the first line of each item, or following the last line of a brief, indicate how the original information was processed. Where no processing indicator is given, the information was summarized or extracted.

Unfamiliar names rendered phonetically or transliterated are enclosed in parentheses. Words or names preceded by a question mark and enclosed in parentheses were not clear in the original but have been supplied as appropriate in context. Other unattributed parenthetical notes within the body of an item originate with the source. Times within items are as given by source.

The contents of this publication in no way represent the policies, views or attitudes of the U.S. Government.

PROCUREMENT OF PUBLICATIONS

JPRS publications may be ordered from the National Technical Information Service (NTIS), Springfield, Virginia 22161. In ordering, it is recommended that the JPRS number, title, date and author, if applicable, of publication be cited.

Current JPRS publications are announced in Government Reports Announcements issued semimonthly by the NTIS, and are listed in the Monthly Catalog of U.S. Government Publications issued by the Superintendent of Documents, U.S. Government Printing Office, Washington, D.C. 20402.

Correspondence pertaining to matters other than procurement may be addressed to Joint Publications Research Service, 1000 North Glebe Road, Arlington, Virginia 22201.

Soviet books and journal articles displaying a copyright notice are reproduced and sold by NTIS with permission of the copyright agency of the Soviet Union. Permission for further reproduction must be obtained from copyright owner.

3 December 1982

USSR REPORT
CYBERNETICS, COMPUTERS AND AUTOMATION TECHNOLOGY

No. 65

CONTENTS

GENERAL

Expanded Computer Production Discussed.....	1
Organization of and Technology for Operation Using Centralized Planning and Introduction of Software in Gosplan Automated Control System for Planning Calculations.....	3
Work of ESSR Academy of Sciences Institute of Cybernetics.....	5
Eliminating Effects of Electromagnetic Radiation.....	7
Improving Management of Plant Computing Centers.....	8

SOFTWARE

Selection of Coding Quality Criterion of Space-Frequency Filters for Coherent Optical Processor.....	14
Standard Subroutine Pack for Working With CAMAC Equipment.....	20
Standard Graph Pack GRAS-Graphical Data Display, Storage and Transmission Devices.....	22
Software for Statistical Analysis of Aerial Photographs of Timber.....	23
Organization of Systems Software of Aerospace Data Processing Complex.....	33
Interactive Printed-Circuit Card Design System.....	39

APPLICATIONS

Dialog With Computer Continues.....	45
Method of Organizing Interactive Mode for Conducting Automated Laboratory General Physics Experiment.....	47
Multiterminal Systems for Academic Research Work at Technical VUZes.....	57

PUBLICATIONS

Table of Contents of Journal 'AVTOMETRIYA', July-August 1982.....	59
---	----

GENERAL

EXPANDED COMPUTER PRODUCTION DISCUSSED

Paris ZERO UN INFORMATIQUE HEBDO in French 6 Sep 82 p 44

[Article by Bohdan O. Szuprowicz]

[Text] By the end of the 11th Five-Year Plan (1981-1985), the countries of the east are planning to double their computer production, raising it to a cumulative total of some \$20 billion. The outcome of a new international computer agreement, this plan was signed at the 34th session of the CEMA [Council for Economic Mutual Assistance].

The program includes about 20 research institutes and development centers, and more than 70 manufacturing plants producing 300 different items in all the CEMA countries, plus Cuba.

Over 46,000 technicians, engineers, researchers, and about 300,000 workers are engaged in this project. This means that in the next few years, the Soviet bloc will produce about 6 times less than IBM, for a value of \$11,500 a year per worker.

Because of the present international situation, the Soviet bloc's computer industry, now more than ever before, sees that its future performance will be based directly on its own present research and development work. Given a scenario of growing east-west tensions, including a restriction on credit and a ban on technology transfer from the west to the countries of the east, this aspect could well become quite important.

Although the other CEMA countries are involved industrially in this program, the Soviet program takes on crucial importance because of the military applications involved. The computer plan includes the manufacturing of medium, large, and very large systems, along with minicomputers, terminals with display screens, and a large number of peripheral items.

This program is being managed by the GKNT [State Committee for Science and Technology] which handles not only its coordination inside the Soviet Union, but also maintains relations with foreign countries in order to follow new technological developments and introduce them into the Soviet Union. This is the agency with which Western companies such as Control Data, CII-HB, Hewlett-Packard, and a good many others sign cooperation agreements in the hope that it will help to open up the Soviet market.

The GKNT has a division, the CTSCA [Main Administration for Computer Technology and Control Systems], which is responsible for computer technologies. It has a special department which indicates the direction research should take.

The CTSCA is headed by Vladimir Aleksandrovich Myasnikov, who is well known from his reports on the Soviet Union's computer policy.

The GKNT operates a scientific and technical library in which foreign publications are catalogued, summarized, and then disseminated, with the assistance of a center headed by Boris Zverkov.

The current computer research and development program is being conducted by institutes attached to the Academy of Sciences, to the academies of the various republics, the ministry of the radio industry, and the MINPRIBOR [Ministry of Instrument Making, Automation Equipment, and Control Systems]. Most of these institutes are located in Moscow, Kiev, Minsk, and Tbilisi.

7679

CSO: 1853/1

ORGANIZATION OF AND TECHNOLOGY FOR OPERATION USING CENTRALIZED PLANNING AND
INTRODUCTION OF SOFTWARE IN GOSPLAN AUTOMATED CONTROL SYSTEM FOR PLANNING
CALCULATIONS

Moscow KLASSIFIKATORY I DOKUMENTY in Russian No 4, Apr 82 pp 1-5

[Article by candidates of technical sciences L.V. Pavel'yeva and N.N. Fedotov,
Main Computer Center, USSR Gosplan; GNITSVOK]

[Excerpts] The planning and use of general-system software when doing planning calculations are associated with the problem of developing a centralized data file for an automated data bank for the automated control system for planning calculations [ASPR]. Development of the data file for the automated data bank used in the ASPR is fraught with a number of difficulties, the main ones of which are the following:

inadequate operating speed and restricted sizes of main and external storage for the base computer of the ASPR (the Yes-1033);

the need to develop for the ASPR an additional structure for the YeS online system and an "OKA" standard control system for the base data [SUBD];

the need to develop general-system data facilities for forming customized nomenclature for users;

the need for an organizational transformation resulting from the growing trend toward distributed data processing required for doing planning calculations.

The main stages in the preparation and processing of data in a unified technological process are as follows:

- 1) description and preparation of input data;
- 2) loading, correcting and fetching data from the primary data bases;
- 3) compiling a word glossary and building the field for input data;
- 4) forming the numeric part of the data file and loading general-system numeric and local data bases;
- 5) logic control and the execution of planning calculations using data from the data file for the automated data bank.

In the main computer of the USSR Gosplan three types of bases have already been developed in minimum configuration and are being operated using the program facilities of the "OKA" SUBD, the OKA Data Transmission System" [SPD-OKA] and a system for machine control of the ASPR glossary. There are two bases for the form of planning documents: with numeric values for economic indicators and their coded symbols (the numeric bases), and also a text base for a general-system ASPR glossary with textual designations for economic indicators and their labeled parts and the corresponding coded designations.

In addition to the above, in the second stage it is necessary to form a base for descriptions of the economic indicators, planning documents, tasks, subsystems, nomenclature and other objects subject to data analysis in the preplanning stage when designing base textual and numeric data files for the automated data bank. This is the so-called ASPR metadata base. The metadata also include tables reflecting the link between the data units and objects described. The organizational-technical measures of the second stage are as follows:

loading and three types of mass correction to the data base whose operations are typical in forming the internal level of the automated data bank and which are executed by the facilities of the SPD-OKA:

three types of individual corrections to the data bases that in the future should be done using the facilities of the TIS-Yes but which at present are done using the facilities of the TIS system on an ICL 4-70 and the main facilities of the SPD-OKA;

passing working arrays from numeric and local textual data using the facilities of the SPD-OKA system;

fetching and passing from the general-system glossary data base and the metadata base using the facilities of the glossary system.

COPYRIGHT: Vsesoyuznyy nauchno-issledovatel'skiy institut tekhnicheskoy informatsii, klassifikatsii i kodirovaniya, 1982

9642

CSO: 1863/343

WORK OF ESSR ACADEMY OF SCIENCES INSTITUTE OF CYBERNETICS

Moscow PRAVDA in Russian 2 Aug 82 p 3

[Article by A. Favorskaya]

[Excerpts] Tallinn--Kharri Tani, deputy director of the Special Design Bureau at the Estonian SSR Academy of Sciences Institute of Cybernetics, took me round the laboratories, where the computers are by no means run of the mill.

Where we are now is a world of microprocessor control systems and minicomputers and microprocessors. And typically, they are not made to show at any exhibition but directly for a specific client. Sometimes they are developed jointly with the client. Things go more rapidly this way: even during the development stage the system can be "tailored" most closely with the conditions in which it will be used. There are plenty of orders for miniaturized computer equipment. For example, for agriculture, an automated control system for drying grain has been developed jointly with the Estonian Research Institute of Agriculture and Land Reclamation where they know the drying technologies "from the inside." And why is development so rapid? Because the anticipated annual saving throughout the republic is close to R1 million. And at the Estonian SSR Academy of Sciences Institute of Physics, laser experiments will also be automatically controlled with a microprocessor developed in the special design bureau.

The special design bureau itself--a four-storey building with laboratories and workshops--did not exist 6 years ago.

The go-ahead has now been given for technology to mass produce microprocessors and also to develop new automation instruments based on microprocessors. But this will require flexible means of automation for designing and fabrication PCB's, hybrid circuits and entire microprocessors. Facilities are required for debugging and setting-up these machines. At present they are in short supply. This new technologic base has enabled the Tallin Special Design Bureau to reduce the time spent on development and produce qualitatively new items and even sometimes produce small batches of them.

Young and talented colleagues have now joined the 30 or so of the republic's best electronics experts who formed the nucleus of this bureau. The special design bureau now has a staff of about 300, almost half of which are candidates of sciences.

Going down the list, we start with the director Kal'yu Leppik, an energetic and skillful organizer who has an economic approach, which is important in development. Indeed, everyone here has his own characteristic creative aspect. Yuriy Ummer and Enn Tal'vis (who worked on the above-mentioned grain drier) are outstanding analysts. Reyn Mikhkel'son, who developed a measuring device for concrete, is just as intelligent and is also known for his "skillful" hands. Kaarel Myartin is recognized among his colleagues as an brilliant programmer. Raul' Rebane has shown himself to be extremely creative when participating in the development of the SM-1800 series-produced computer. And without Osval'd Sulu the chief engineer, it seems that the special design bureau would cease to be.

Today, this collective is involved in six all-union, goal-oriented programs, from automated control of technological processes and scientific experimentation with the use of microprocessors to microprocessor applications in new robot technology. Some work is being done for CEMA.

9642

CSO: 1863/350

ELIMINATING EFFECTS OF ELECTROMAGNETIC RADIATION

Riga SOVETSKAYA LATVIYA in Russian 18 Aug 82 p 1

[Unattributed article: "To Eliminate 'Glitches' in the Computer"]

[Text] The warning signal "Unreliable Operation" is appearing less often on the displays screens of Soviet-produced computers. Operating stability in computer equipment is being helped by a set of equipment developed by specialists at the Vilnius "Sigma" Association.

Not only the medium in which it is operating but also the electromagnetic pulses and oscillation created by the computer itself exert a definite effect on electronic equipment. Passing into the "memory" of the machines, they distort responses to problems being solved. The new set of instruments "damps" this noise, and the degree of protection afforded for electronic equipment by the "Sigma" apparatus has been improved by a factor of better than 10.

9642

CSO: 1863/350

IMPROVING MANAGEMENT OF PLANT COMPUTING CENTERS

Kiev EKONOMIKA SOVETSKOY UKRAINY in Russian No 7, Jul 82 pp 77-80

[Article by N. Leybovich, docent, candidate of economic sciences, Dnepropetrovsk: "On Improvement of Management of Plant Computing Centers"]

[Text] Computing and data processing centers (IVTs's) are essentially service departments for processing data with individual rights and obligations. For purposes of improving the efficiency of functioning of plant IVTs's it would be advisable within the next 2 or 3 years to convert them to in-house cost accounting and to create the possibility of serving other enterprises and organizations on a cost accounting basis. The changeover of IVTs's of industrial enterprises and associations to cost accounting is a necessary measure under conditions of the formation of multiple-user computing centers and collective-use computing centers and the creation of a network of computing centers.

The introduction of cost accounting at an IVTs presupposes a high level of organization of planning and accounting and analysis of all aspects of operations and of all technical and economic indicators of the operation of the IVTs and its structural subdivisions, and the regulation of cost accounting interrelations with other sections and departments of the plant administration. For the purpose of converting plant IVTs's to cost accounting for the purpose of improving the efficiency of their operation, it would be helpful to improve accounting, planning and analysis of the operations of the center, to organize the development of time and material input norms for planning and solving ASUP [automated enterprise management system] problems, to perform economic studies of IVTs expenses, to create a methodology for economic calculations of the operating indicators of an IVTs, and to form an economic data basis for organizing an IVTs ASU [automated control system], having improved the level of economics work within the IVTs. It would be necessary to isolate the IVTs from the plant administration structure and to regard it as a self-supporting department for planning and automated data processing, having accordingly restructured the planning and recording of its technical and economic indicators.

Since an IVTs is intended primarily to solve economic problems and to improve control of production and the planning and accounting procedure, then, as with other economic departments and laboratories of the plant, it is logical to place it under the plant's chief economist. An economics sector (two or three people specializing in economics and organization of computing centers) is required at an IVTs with the functions of setting norms for labor and material resources, operations

planning and controlling computing processes, current and longterm planning of technical and economic indicators for the operation of the IVTs and bookkeeping and statistical reporting, of analyzing the work time and volume of IVTs subdivisions, and of making an economic analysis of the results of the work of IVTs sectors and of estimates of the efficiency of solving ASUP problems. This in turn will make it possible to accumulate statistical data for the purpose of developing standards and to improve calculation of the cost of solving individual problems.

It is a good idea to keep records with an indication of the task code for the work time of computers and workers (analysts, programmers and operators), as well as for the amount of data processed in appropriate units of measurement. Then the work time and data processing costs per production section can be related to the solution of individual problems and it will be possible to determine and analyze data processing costs task by task. In keeping records of the working time of a computer it is proposed to differentiate calendar time into days of rest, holidays, non-working hours and duty time, which is divided into the time for modernization and repairs with the power switched on and the time of the working switched-on state, which consists of useful production time and non-productive time. Non-productive time is spent on preventive maintenance and modernization of computers and includes computer idle time resulting from equipment malfunctions, errors and failures, lack of power, errors of operators and programmers, lack of work, tardy presentation of source data, tardy or poor-quality preparation of data media, etc.

Useful time consists of effective time (solving problems, standard time for debugging programs, training personnel on the computer) and ineffective time (above-norm debugging time; redoing work because of errors and rejects; repeated solution of problems resulting from errors, malfunctions and failures; an operator's or programmer's waiting and thinking over the work run on the computer; and nonoverlapping time for preparatory and final and auxiliary manual operations). In addition, it is necessary to keep records by automated means of nonoverlapping time for input/output and accessing external storages and the time for operation of the central processor in the single-program and multiprogram modes (task by task) and the work time of the operating system.

For the possibility of calculating the cost of solving problems it would be helpful to keep records of labor, material and capital input by item of expenditure, by production section and by ASUP task.

The conversion of IVTs's to cost accounting should be preceded by working out wholesale prices (rates) for IVTs services. For this it would be necessary to make estimates of standard service time and hourly costs for the operation of individual computer equipment. It should be mentioned that the fixed capital of a plant IVTs equals more than 1 million rubles and the cost of data processing 500,000 to 800,000 rubles per annum.

It has been established as the result of preliminary research that the structure of the cost of data processing at a plant computing and data processing center (without general-plant expenditures) looks as follows (percentages):

A. By Item of Expenditure

Basic and supplementary wages of production personnel with deductions for social security	39.3
Electric power for production purposes	2.0
Routine repairs and maintenance of production equipment	12.2
Depreciation of production equipment	34.0
Materials	1.0
Wages of administrative and servicing personnel with deductions for social security	7.2
Travel allowance expenditures	0.5
Routine repairs and maintenance of buildings, structures and inventory	1.7
Depreciation of buildings, structures and inventory	1.0
Wear of low-value and rapidly wearing items	0.1
Cost of training and improving skills of personnel	0.1
Others	0.9

B. By Production Section

Analysis and programming	17.0
Receiving primary documents	3.2
Processing on keyboard computers	6.6
Transfer to and preparation of data on punched-card computers	10.5
Punching, sorting and tabulating	12.6
Operation of "Minsk-32" computer	18.5
Operation of "YeS-1022" computer	25.2
Checking and output of data	6.4

With regard to items of expenditure all costs, besides materials, electric power for production purposes and piece-rate wages of operators, are hypothetically constant. With regard to combined production sections, direct costs are related directly to production sections and indirect costs are distributed proportionally to direct.

Analysis of these costs makes it possible to determine ways of increasing the operating efficiency of IVTs's. Knowing hourly costs and the time required for data processing in individual production sections and with individual units in solving ASU problems, it is possible to determine the cost of solving problems. If the profitability of data processing is established (let us say, 10 percent of the standard cost of data processing), then it is possible to determine wholesale prices for an hour of use of individual pieces of equipment and for solving various problems, as well as surcharges and discounts on them for changes in the time required for and quality of data processing.

For estimating the amount of data in solving ASUP problems it is necessary to know the hourly mean throughput of the computer in output data characters (by type of problem in accordance with the complexity group and coefficients for the ratio of the volume of output and input data) and the standard time for solving problems. The sum of products of the hourly mean throughput of the computer by the standard problem solving time determines the planned number of characters to be output from the computer to an output unit per month or year.

Table 1. Calculation of Cost of Computing Work by IVTs Section

<u>Name of section</u>	<u>Receipt and re-cording of primary documents, man-h</u>	<u>Keyboard computer</u>	<u>Punchcard computer</u>	<u>Tabulation</u>	<u>Checking and out-put</u>	<u>Algorithm-ization and pro-gramming, man-h</u>	<u>"Minsk-32" computer</u>	<u>"YeS-1022" computer</u>
		<u>Rated output per man-hour</u>			<u>Man-hours</u>		<u>Machine-hours</u>	
Volume of work	12240	22440	22440	18360	20400	51000	3800	3800
Total cost, rubles	10121	20522	32425	38940	20092	52464	56545	78208
Cost of one hour, rubles	0.83	0.91	1.44	2.12	0.98	1.03	14.88	20.58

Plan and report calculations of the cost of processing and solving problems make it possible more soundly to analyze and establish wholesale prices (as well as surcharges and discounts on them) for IVTs services elsewhere, as well as estimate intraplant prices for intraplant accounting. In addition, they will make it possible more soundly to determine the economic efficiency of processing and of solving individual problems and of the entire automated production control system at each enterprise, and to improve the effectiveness of socialist competition for better technical and economic indicators for the work of IVTs subdivisions.

In order to give grounds for the planning of production output, labor productivity and the cost of processing and solving problems, a system of cost standards is required--for labor, computer, material, energy, etc. costs, involved in processing a specific amount of raw data. Standards are needed for elements of the work time of various computers and computing center workers, including components of computer time for solving problems and debugging programs and for formulating problems and writing programs.

Solutions are required to methodological problems relating to improving determination of the volume and complexity of data to be processed; computing center work volume indicators; regulating the work of ASU developers (formulators and programmers); calculating the cost of developing and solving individual ASU problems and processing a million characters of raw, as well as of output, data; determining wholesale and estimated prices (as well as surcharges and discounts on them) for computing center services; determining computer center output-capital ratio and profitability indicators with possible differentiation by production section and by individual task; the planned and actual economic efficiency of using computing centers and ASU's; and converting all computing centers to total or in-house cost accounting, having provided them with a reliable system of planning, keeping records of and analyzing results and the operating indicators of computing centers. Conversion of an IVTs to cost accounting should be preceded by development of a bonus system to provide an incentive for the economically more efficient solution of ASU problems and use of IVTs material, labor, money and information resources. It is possible to recommend a method of awarding bonuses for the development and introduction of ASUP tasks with the observance of time deadlines and standards (in man-hours) as a function of the economic efficiency of ASUP tasks designed: for formulation of the problem--10 percent of its actual annual saving upon condition of integrated data processing. The bonus is distributed among coworkers in proportion to the standard wages of problem formulators (from departments and divisions of the plant administration and IVTs), taking into account the time spent on participating in the work. And for programming a problem at the state of the art, ensuring a minimum requirement of machine time for solving the problem--5 percent of its actual annual saving with a similar procedure for distributing the total bonus. And for introducing and mastering a task in the IVTs's operation--5 percent of its actual annual saving with similar distribution of the total bonus.

By the saving is meant the total reduction in costs per ruble of commodity production and increase in profit from growth in volume and realization (after subtracting the cost of developing the problem and of automated data processing) gained in connection with the introduction of a specific ASUP task at a specific enterprise per year. Bonuses should be paid to operators and other IVTs operating

staff upon condition of reduction in the cost of processing a unit of information, taking into account timeliness, reliability and freedom from errors in fulfilling the computing work plan; and to personnel servicing electronic and electromechanical equipment and to attendants and repair personnel--for freedom from failure and reliable performance and for reduction of idle time of individual computers associated with equipment malfunctions and repairs.

A system of penalty sanctions for parties guilty of violations in favor of the victims should be conducive to improving the organization of document turnover and to improving the efficiency of control and the effectiveness of automated and mechanized data processing. The penalty system must complement the bonus system and be accomplished at the expense of total bonuses to coworkers.

A sound system of technical and economic indicators and clear, simple and unified accounting of the work of differentiated production sections and of data processing costs for these sections, and planning and accounting of the number, wages and bonuses of IVTs personnel together with an analysis of economic activity and a determination of the economic efficiency of solving individual problems will create the data and economic conditions for converting an IVTs to intraplant cost accounting and the basis of the data and economic support for an IVTs ASU. Added costs for carrying out the accounting process and for calculating plan figures will be totally reimbursed on account of reduction of the IVTs operating costs and the more efficient use of computers in optimizing the solution of ASUP and IVTs ASU planning and accounting problems on the basis of integrated data processing. It would be helpful to automate accounting and planning and analysis of the work of plant IVTs's.

The creation of a system for planning, keeping account of and making an economic analysis of all aspects of the economic activity of an IVTs is the foundation for organizing an IVTs ASU. The creation of an IVTs ASU represents the logical conclusion of the combination of work on automating the control of production (organization) and technological processes. Already at the present time it is possible to create the data, organization and economic support for this purpose.

And so, for purposes of improving the economic efficiency of an IVTs and the level of control of these subdivisions it is necessary to convert the IVTs's of industrial enterprises and associations to in-house cost accounting, to regard them as service departments for processing economic data and to raise the level of economic work at IVTs's.

It is necessary to continue scientific research on improvement of control, planning, keeping accounts and analysis of economic indicators of the work of computing and data processing centers for purposes of improving the efficiency of automated processing of economic data.

COPYRIGHT: Izdatel'stvo "Radyans'ka Ukrayina", "Ekonomika Sovetskoy Ukrainy", 1977

8831

CSO: 1863/10

SELECTION OF CODING QUALITY CRITERION OF SPACE-FREQUENCY FILTERS FOR COHERENT OPTICAL PROCESSOR

Leningrad IZVESTIYA VYSSHIKH UCHEBNYKH ZAVEDENIY: ~~PR~~IBOROSTROYENIYE in Russian Vol 25, No 8, Aug 82 (manuscript received 21 Oct 80) pp 54-58

[Article by Ye. F. Ochín, Leningrad Institute of Precision Mechanics and Optics]

[Text] Separation of an infinite set of transfer functions for a coherent optical processor into a finite number of classes is proposed. A problem of finding the best coding algorithm from a set of known algorithms is solved for each class. The problem of synthesis of space-frequency filters is reduced to determination of the class of the transfer function and to use of the corresponding coding algorithm.

Let the complex transfer function of a space-frequency filter of a coherent optical processor be given in the form

$$H(\xi, \eta) = \sum_m \sum_n a(\xi_n, \eta_m) \exp[i\varphi(\xi_n, \eta_m)] \operatorname{rect} \frac{\xi_n - \xi}{\Delta\xi} \operatorname{rect} \frac{\eta_m - \eta}{\Delta\eta}, \quad (1)$$

where ξ and η are space-frequency coordinates, $\Delta\xi$ and $\Delta\eta$ are the quantification steps, a and φ are the amplitude and phase modulation functions, $-\xi_{\max}/\Delta\xi \leq n \leq \xi_{\max}/\Delta\xi$; $-\eta_{\max}/\Delta\eta \leq m \leq \eta_{\max}/\Delta\eta$; $\xi_n = n\Delta\xi$; $\eta_m = m\Delta\eta$; $|a(\xi_n, \eta_m)| \leq 1$.

The pulse response of the filter, which is Fourier transform (1), is written in the form

$$I(x, y) = \sum_m \sum_n a(x_n, y_m) \exp[i\varphi(x_n, y_m)] \operatorname{rect} \frac{x_n - x}{\Delta x} \operatorname{rect} \frac{y_m - y}{\Delta y}, \quad (2)$$

where x and y are the coordinates of the output plane of the processor, $\Delta x = 1/2\xi_{\max}$, $\Delta y = 1/2\eta_{\max}$, α and φ are the amplitude and phase of filter response, $-x_{\max}/\Delta x \leq n \leq x_{\max}/\Delta x$, $-y_{\max}/\Delta y \leq m \leq y_{\max}/\Delta y$; $x_{\max} = 1/2\Delta\xi$; $y_{\max} = 1/2\Delta\eta$.

As a result of coding, the fragments of the binary or multilevel transparency are set into agreement to each reference (m, n) of function (1). Using the decoding procedure, some complex number, which is an approximate digital

representation of the precise coded reference of the transfer function, is set into agreement to each fragment of the transparency. The result of coding-decoding of function (1) can be represented in the following manner:

$$\tilde{H}(\xi, \eta) = \sum_m \sum_n \{a(\xi_n, \eta_m) \exp[i\varphi(\xi_n, \eta_m)] + \varepsilon(\xi_n, \eta_m)\} \times \\ \times \operatorname{rect} \frac{\xi_n - \xi}{\Delta \xi} \operatorname{rect} \frac{\eta_m - \eta}{\Delta \eta}, \quad (3)$$

where $\varepsilon(\xi_n, \eta_m)$ is the coding error.

The pulse response of the encoded filter assumes the form

$$\tilde{I}(x, y) = \sum_m \sum_n \{a(x_n, y_m) \exp[i\varphi(x_n, y_m)] + Q(x_n, y_m)\} \times \\ \times \operatorname{rect} \frac{x_n - x}{\Delta x} \operatorname{rect} \frac{y_m - y}{\Delta y}, \quad (4)$$

where $Q(x_n, y_m)$ are the references of the Fourier transform of the coding error.

The problem of selecting the coding quality criterion of functions of type (1) was first considered in [1]. Based on the given function (2), its Fourier transform (1) was calculated, which was subjected to the coding-decoding procedure. Using the reciprocal Fourier transform from function (3), conversion to function (4) was made and the following value was selected as the coding quality criterion

$$\rho_1 = \frac{1}{1 x_{\max} y_{\max}} \int_{-x_{\max}}^{x_{\max}} \int_{-y_{\max}}^{y_{\max}} ||I(x, y) - \tilde{I}(x, y)||^2 dx dy. \quad (5)$$

Further development of the problem of selecting the coding quality criterion was given in [2-4]. It was suggested in the given paper that the functional of the following type be used as the coding quality criterion

$$\rho_2 = \frac{1}{1 x_{\max} y_{\max}} \int_{-x_{\max}}^{x_{\max}} \int_{-y_{\max}}^{y_{\max}} ||H(\xi, \eta) - \tilde{H}(\xi, \eta)||^2 d\xi d\eta. \quad (6)$$

If transfer function (1) is given, the use of criterion (6) does not require the use of the Fourier transform. If the pulse response (2) is given, the Fourier transform is used once. Let us show that analyses (5) and (6) are equivalent. By substituting (2) and (4) into (5), we find

$$\rho_1 = \frac{\Delta x \Delta y}{1 x_{\max} y_{\max}} \sum_m \sum_n |Q(x_n, y_m)|. \quad (7)$$

By substituting (1) and (3) into (6), we find

$$\phi_2 = \frac{\Delta \xi \Delta \eta}{\xi_{\max} \eta_{\max}} \sum_m \sum_n |e(\xi_m, \eta_m)|. \quad (8)$$

$Q(\xi_n, \eta_m)$ and $E(\xi_n, \eta_m)$ are linked by a discrete Fourier transform; therefore, one can write

$$\sum_m \sum_n |Q(x_n, y_m)| = \sum_m \sum_n |e(\xi_n, \eta_m)|.$$

Since $\Delta x = 1/2\xi_{\max}$, $\Delta y = 1/2\eta_{\max}$, $x_{\max} = 1/2\Delta\xi$, $y_{\max} = 1/2\Delta\eta$, then $\frac{\Delta x \Delta y}{x_{\max} y_{\max}} = \frac{\Delta \xi \Delta \eta}{\xi_{\max} \eta_{\max}}$, and accordingly $\psi_1 = \psi_2$.

It follows from expression (8) that the mean coding error is not dependent on the mutual arrangement of the encoded points. Since transfer function (1) has

$N = \frac{\xi_{\max} \eta_{\max}}{\Delta \xi \Delta \eta}$, one can form $N!$ different transfer functions having identical mean coding errors by permutations. The set of indicated transfer functions can be described naturally by some statistical characteristics, for example, by the two-dimensional differential law of distribution of the modulus and independent variable of the references of function (1). Let us assume that the indicated distribution law has independent components of distribution of the modulus and independent variable, i.e.,

$$I(r, \varphi) = I(r)I(\varphi), \quad (9)$$

where r and φ are the modulus and independent variable of a complex number, which is the reference of function (1).

We note the most important practical classes from an infinite set of distribution laws (9) (see table).

$I(r)$	$I(\varphi)$		
	$\delta(\varphi)$	$[\delta(\varphi) + \delta(\varphi + \pi)]/2$	$\frac{1}{2\pi} \text{rect}(\varphi/2\pi)$
$\delta(r - 1)$	$I(\xi, \eta) = 1$ (рис. 1) (1)	Бинарно-фазовая (рис. 2) (2)	Равномерно-фазовая (рис. 3) (3)
$[\delta(r) + \delta(r - 1)]/2$	Бинарно-амплитудная (рис. 4) (4)	Бинарно-комплексная (рис. 5) (5)	Бинарно-амплитудная равномерно-фазовая (рис. 6) (6)
$\text{rect}(r - 0.5)$	Равномерно-амплитудная (рис. 7) (7)	Равномерно-амплитудная бинарно-фазовая (рис. 8) (8)	Равномерно-комплексная (рис. 9) (9)

Note. δ is a Dirac delta-function.

Key:

- | | |
|----------------------------|--------------------------------|
| 1. Figure 1 | 3. Uniform-phase (Figure 3) |
| 2. Binary-phase (Figure 2) | 4. Binary-amplitude (Figure 4) |

[Key continued on following page]

[Key continued from preceding page]:

6. Binary-complex (Figure 5)
- Binary-amplitude uniform-phase (Figure 6)
7. Uniform-amplitude (Figure 7)
8. Uniform-amplitude binary-phase (Figure 8)
9. Uniform-complex (Figure 9)



Figure 1



Figure 2



Figure 3



Figure 4



Figure 5

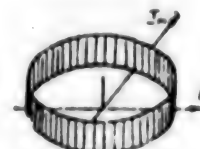


Figure 6

The distributions presented in Figures 1-9 correspond to special cases of transfer functions of more general classes. Thus, for example, the distributions shown in Figures 1, 4 and 7 are related to the class of non-negative amplitude transfer functions, the general form of which can be represented in the following manner: $H(\xi, \eta) = a(\xi, \eta)$, where $0 \leq a(\xi, \eta) \leq 1$.

The distributions presented in Figures 2, 5 and 8 are related to the class of alternating amplitude transfer functions of type $H(\xi, \eta) = a(\xi, \eta)$, where $|a(\xi, \eta)| \leq 1$.

Distributions 1, 2 and 3 belong to the class of phase functions of type $H(\xi, \eta) = \exp[i\varphi(\xi, \eta)]$.

The distributions presented in Figures 6 and 9 belong to the class of complex functions of general form (1)



Figure 7



Figure 8



Figure 9

The problem of finding the best coding algorithm from a set of known algorithms is solved for each class of transfer functions. One or several realizations of random process (9), for which the problem of selecting the coding algorithm by the minimum error criterion (6), is generated by using the random number sensor

according to the expressions presented in the table. In sum, the coding problem reduces to determination of the class of the transfer function and to use of the corresponding algorithm.

Let us select the distribution shown in Figure 9 for comparative analysis of the coding quality of different algorithms in the more general case when the class of the transfer function is unknown. Let us show that by introducing different weight functions from a distribution of form

$$f(r, \varphi) = \frac{1}{2\pi} \text{rect}(r - 0.5) \text{rect}(\varphi/2\pi), \quad (10)$$

one can find any class of distributions since the amplitude component of functions (1) is a limited value, i.e.,

An infinite set of realizations of function (1) corresponds to distribution (10). The resulting specific realizations is related to the procedure of generation of random numbers. Since distribution (10) has a comparatively simple form, then one can suggest a set of specific functions (1) having distribution (10), for example:

$$H(\xi_n, \eta_m) = \left[\left(\frac{\xi_n}{\xi_{\max}} \right)^2 + \left(\frac{\eta_m}{\eta_{\max}} \right)^2 \right]^{1/2} \exp \{ 2\pi i \lg^{-1}(\eta_m/\xi_n) \} \times \\ \times \text{rect} \frac{\xi_n - \xi_0}{\Delta \xi} \text{rect} \frac{\eta_m - \eta_0}{\Delta \eta}. \quad (11)$$

It follows from expression (11) that the modulus of the maximum error of being given the transfer function is $\epsilon_{\max} = \frac{1}{2} \sqrt{\Delta \xi^2 + \Delta \eta^2}$.

Let us assume that the coding error is equal to zero if its modulus does not exceed the value ϵ_{\max} , i.e.,

$$\epsilon(\xi_n, \eta_m) = \begin{cases} \epsilon(\xi_n, \eta_m) & \text{at } |\epsilon(\xi_n, \eta_m)| > \epsilon_{\max} \\ 0 & \text{at } |\epsilon(\xi_n, \eta_m)| \leq \epsilon_{\max} \end{cases} \quad (12)$$

Let us return to the procedure of generation of universal differing structures represented in the framework of a single fragment of an encoded transparency and that does not contradict the coding algorithm: $S = S_0, S_1, \dots, S_k$. A complex number C_j : $C = C_0, C_1, \dots, C_k$, is assigned to each structure of S_j by using the coding procedure. If $|C_j| < 1$, then at least one reference with number (n, m) , for which $|H(\xi_n, \eta_m) - C_j| \leq \epsilon_{\max}$, i.e., a precisely given coding reference with regard to (12), is always found among references (11). One can show that the corresponding indices are determined in the following manner:

$$m = \text{int} \left[\text{Im}(C_j) \frac{\epsilon_{\max}}{\Delta \eta} + 0.5 \right], \quad | \\ n = \text{int} \left[\text{Re}(C_j) \frac{\epsilon_{\max}}{\Delta \xi} + 0.5 \right]. \quad (13)$$

A value of $\epsilon(\xi_n, \eta_m) = 0$ is assigned to each reference (8) with indices from (13). Let us find the closest reference (m_0, n_0) , equal to zero for each of the remaining references (m, n) and let us perform the operation:

$$\epsilon(\xi_n, \eta_m) = \max[(m - m_0)^2 + (n - n_0)^2]^{1/2}.$$

Thus, the procedure of determining references $\xi(\xi_n, \eta_m)$ for transfer functions having distribution (10) is not dependent on a specific type of function (1) and one can compare the different coding algorithms by using analysis (8). One can show that a similar approach can also be used for distributions differing from (10).

BIBLIOGRAPHY

1. Gabel, R. A. and B. Lin, "Minimization of Reconstruction Errors with Computer-Generated Binary Holograms," APPLIED OPTICS, Vol 9, No 5, 1970.
2. Ransom, P. L., "Synthesis of Complex Optical Wavefronts," APPLIED OPTICS, Vol 11, No 11, 1972.
3. Rowers, R. C. and J. W. Goodman, "Error Rates in Computer-Generated Holographic Memories," APPLIED OPTICS, Vol 14, No 7, 1975.
4. Buchlew, I. and N. C. Gallagher, "Comprehensive Error Models and a Comparative Study of Some Detour-Phase Holograms," APPLIED OPTICS, Vol 18, No 16, 1979.

COPYRIGHT: "Izvestiya vuzov SSSR - Priborostroyeniye", 1982

6521

CSO: 8144/0097

STANDARD SUBROUTINE PACK FOR WORKING WITH CAMAC EQUIPMENT

Novosibirsk AVTOMETRIYA in Russian No 4, Jul-Aug 82 (manuscript received 11 Dec 81) pp 17-23

[Article by O. V. V'yushin and P. L. Khrapkin, Novosobirsk]

[Excerpts] 1. Introduction. General description of subroutine pack. The operating experience of scientific centers shows that a large part of applied programs is written in FORTRAN language. Compilation of the experiment control and data processing program in this language for experimenters is usually simpler and faster than in Assembler language. However, FORTRAN syntax has no standard devices for operating with CAMAC equipment. An applied subroutine pack written in Assembler language must be developed to guarantee the capability of the use of CAMAC by a wide range of experimenters. The called pack subroutines are usually included in the user program at the Assembly stage. The ESONE/SR/01 standard "SUBROUTINES FORCAMAC" was taken as the basis in writing the pack [1].

The subroutine pack is designed for working on the Elektronika-60 or SM-3 microcomputers with RSX-11 multiprogram real-time operating system. Type CC-11, M400, SM-3, K-16 and E-60 CAMAC controllers were used. With slight modifications, the pack can be used with other types of controllers, on other computers of the SM series, the Elektronika series or the PDP-11 and with other operating systems (see section 7 for more details).

The pack subroutines are not oriented toward servicing of some specific equipment modules. They permit one to work with any devices in the CAMAC standard since they realize themselves the requirements of this standard: execution of CAMAC instructions, generation of general-control signals (Z, C, I) and data exchange modes provided by the standard.

BIBLIOGRAPHY

1. SUBROUTINES FOR CAMAC, ESONE/SR/01, September 1978.
2. Kazakova, N. A. and Ye. V. Pankrats, "Realization of IML Intermediate Level Language on SM-3 Type Computers," AVTOMETRIYA, No 3, 1980.

3. "The Definition of IML, A Language for Use in CAMAC System," ESONE/IML/01, ESONE SECRETARIAT, October 1974.
4. Sendall, D. M., "Minicomputer Software for High-Energy Physics Experiments", Proceedings of the 1978 CERN School of Computing, June 1978.

COPYRIGHT: Izdatel'stvo "Nauka", "Avtometriya", 1982

6521

CSO: 1863/2

STANDARD GRAPH PACK GRAS-GRAPHICAL DATA DISPLAY, STORAGE AND TRANSMISSION
DEVICES

Novosibirsk AVTOMETRIYA in Russian No 4, Jul-Aug 82 (manuscript received
29 Jun 81) pp 23-28

[Article by M. A. Kurilov, V. V. Manako, A. I. Nikitin and I. V. Chichkan',
Kiev]

[Excerpts] The standard graph pack GRAS (graph standard) [2-4], which satisfies international recommendations on standardization ACM/SIGGRAPH [5, 6], has been developed at the Institute of Cybernetics, Ukrainian SSR Academy of Sciences.

The structure and functional capabilities of GRAS in display, storage and transmission of graphical data are considered in the article.

Practical realization of the GRAS PDFO [pseudodisplay exchange file] is accomplished within the framework of developing general software of the computer center network of the Ukrainian SSR Academy of Sciences.

Conclusions. The GRAS is now the nucleus of the graph pack of a scientific research hardware-software complex for modelling of the SAPR [automated design system], Institute of Cybernetics, Ukrainian SSR Academy of Sciences, and of the subroutine pack for graphical display of the results of processing scientific information at computer centers of the Ukrainian SSR Academy of Sciences.

COPYRIGHT: Izdatel'stvo "Nauka", "Avtometriya", 1982

6521

CSO: 1863/2

SOFTWARE FOR STATISTICAL ANALYSIS OF AERIAL PHOTOGRAPHS OF TIMBER

Novosibirsk AVTOMETRIYA in Russian No 4, Jul-Aug 82 (manuscript received 18 Jan 82) pp 34-40

[Article by V. A. Ivanov and G. A. Ivanchenko, Novosibirsk]

[Text] A number of problems solved on the basis of an automated image processing complex [1], realized at the IAiE [Institute of Automation and Electrometry], Siberian Department, USSR Academy of Sciences, is related to study and classification of forest sections from aerial photographs. Problems of algorithmic support and software for statistical analysis of images of forest tracks are considered in the given paper.

The results of [2, 3] on the geometric probabilities and point processes were used in development of the algorithms. The extensive bibliography available in the indicated monographs does not permit one to present a detailed list of the literature.

The software considered in the paper can conditionally be divided into two groups. The first group includes programs that operate directly with the image and the second group includes programs that operate with data files on disk units produced by programs of the first group.

Programs of the first group. The LESK program. One of the natural methods of checking the random nature of the distribution of objects on a surface is to check the agreement of the sample distribution to Poisson distribution

$$P(n, \lambda) = (\lambda^n/n!)e^{-\lambda}, \quad (1)$$

where n is the number of objects on a fixed area and λ is the density of objects.

When plotting the sample distribution, one must "adjust" the parameters, which is done in the dialogue mode with visualization of the intermediate phases on the television monitor screen. To do this, the most "typical" fragment of the image (3 X 3 mm) of the section of aerial photograph to be processed is selected; the level of density is given (by the histogram), the number of crowns on the fragment is given and the average area of the tree crown (unit of biomass) is determined; the total number of units of biomass required to select the size of the fixed area and the quantification levels is found on the section to be processed.

The program output data are the distribution of the number of sites with n points, the total area of the crowns and the number of "average" trees.

The LESKRI program. The theory of intersection of random straight lines with nonintersecting objects on a plane is used in applications to ecology [2]. The objects (tree crowns) intersect the secant (a straight line drawn through the section to be investigated) at intervals of length l_0 and the background at intervals of length l_f . The program for the section of image with the given number of intersecting straight lines plus, the distribution of the intervals of the object and the background which can be used in analyzing the average diameters of the crowns, the distances between trees and the density of objects (if the distribution of the crown diameters is known).

Programs for formation of point fields. The following three programs are intended for determination and finding the coordinates of the centers of crowns and their areas with the data recorded in a disk unit file. The characteristics of the images of forest sections on aerial photographs include possible and significant closing of crowns, in variations of the density of the crowns and background (even in small sections), low resolution and high noise level. These characteristics do not permit full automation of the process of determination of crowns and analysis of their parameters, which results in the need to use the dialogue mode.

The LESRT program. For a given section of forest image, the program permits one to find in dialogue mode the centers of tree crowns. The next fragment of the section to be processed is summoned to the TV monitor screen; the centers of the tree crowns, the coordinates of which are entered in the buffer file, are noted with a marker. After passing over the entire section of the image, the coordinates and data on the photograph are recorded in the disk unit file. The program is used in the most complex cases if there is significant closing of crowns and low levels of density of crowns and background.

The LESART program. The program accomplishes semiautomatic separation of crowns and producing analyses of the coordinates of the "centers of gravity" (TsT) and areas. Division of the image into two classes (objects and background) according to a standard is used to detect the crowns. Sample values of the mean density $\bar{\tau}$ and its dispersion S^2 on an elementary fragment (EF) (with minimum dimension of 100 X 100 m and maximum number of points 20 X 20) are used as the feature in division into classes. The values of $\bar{\tau}$ and S^2 of the elementary fragment indicated by a marker on the image brought to the TV monitor screen are taken as the reference values $\bar{\tau}_e$ and S_e^2 .

The following approximation is used as the criterion for division of the fragment to be processed into two classes by mean values [4]

$$\tau - \tau_e > t_{1-\alpha}^v [(S^2 + S_e^2)^{1/2} / \sqrt{N}],$$

where $v = (N + 1) [(S^2 + S_e^2) / (S^4 + S_e^4)] - 2$ and N is the volume of the sample. This division of the fragment into two classes takes into account to some degree the variation of density, unlike simple division by level. The elementary fragments belonging to the objects are then grouped by the threshold distance

from the "center of gravity" of the group. The grouping permits one to separate closed crowns in many cases since the contours in the first approximation can be represented by a circle. The "center of gravity" of the group is taken as the center of the crown and the number of elementary fragments in the group is taken as the area. The resulting centers of the crowns are retrieved (in the form of marks) to the initial image as a verification. If there are significant errors, the operator can repeat processing of the fragment by changing the parameters or he can use the LESK program for processing. If the result is satisfactory, the data are entered in the buffer file and one goes on to the next fragment. The data on the photograph (the number of the photograph, size of the section and number of objects), the coordinates and the areas of the objects are entered in the disk unit file after the given section is processed.

The LESARP program. The program for the density level (given or determined by the density histogram) distinguishes closed objects and counts their "centers of gravity" and the area on the fragment of a 3 X 3 mm image with number of points of 256 X 256. Data on the fragments are entered in the buffer and are recorded in the disk unit file after the entire section has been processed. The program is applicable to images with nonintersecting objects (distinguished by the level of density with small errors).

Programs of the second group. The programs of this group operate with data files that represent files of the coordinates of the centers of crowns produced from images of forest sections.

The LESRRN program. Users are primarily interested in the question of whether the arrangement of points on the section of forest to be studied is random. One of the possible methods is to check the agreement of the sample distribution of distances to the nearest neighbor with theoretical distribution of the distances of the random field of points generated by the Poisson process, which has the form [2]

$$P(r) = 2\pi\lambda r e^{-\pi\lambda r^2}, \quad (2)$$

where λ is the density of points on the section to be considered.

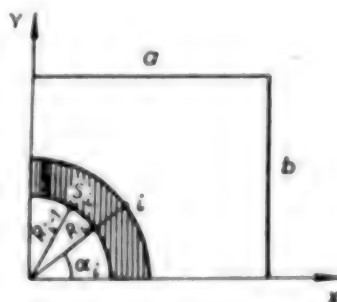


Figure 1. Area S_i and Angle α_i Corresponding to Point i

The program plots the distribution indicated above from the data on the coordinates of points read from the disk unit file and finds the theoretical frequencies on the basis of relation (2). The value of χ^2 is considered to verify the agreement of the sample and theoretical distributions. Moreover, the mean value of \bar{r} and the mean square deviation from the average S_r^2 are calculated.

Investigation of the fields of the centers of crowns of a natural forest showed that the real sample distributions are not described by (2) (see Figure 4), which is related to the finite dimension of the crowns for a mature forest.

The PUASON program. A check of the random nature of arrangement of the centers of objects is less sensitive to the size of the crown if the distribution of areas related to the object is considered.

Sample distributions of the areas and angles are plotted for the points randomly arranged on a rectangular site with sides a and b . The points are first ordered by distance from the beginning of the coordinates to point 1, so that $R_{j-1} \leq R_j$. Area S_j and angle α_j shown in Figure 1 are related to each point 1.

The distributions of the areas and angles for a field of N points in a rectangular zone have the form

$$f(S) = (1/S_{cp}) e^{-S/S_{cp}} = \lambda e^{-\lambda S}, \quad (3)$$

where $S_{gr} = ab/N = 1/\lambda$ and

$$\begin{aligned} \varphi(\alpha) &= (a/2b) \operatorname{tg} \alpha, \quad 0 \leq \alpha \leq \operatorname{arctg}(b/a), \\ \varphi(\alpha) &= b/(2a \operatorname{tg} \alpha), \quad \operatorname{arctg}(b/a) < \alpha \leq \pi/2. \end{aligned} \quad (4)$$

The χ^2 criterion is used to check the agreement of the sample distributions to (3) and (4). The program output data are the sample and theoretical distributions of the areas and angles and the values χ_f^2 and χ^2 .

The VOPLS program. Latticed models are used extensively in applied problems of geology, ecology and so on [3, 5]. Voronny's polygons, which permit one to achieve a number of distributions, are best well known. The algorithms proposed by some authors to plot Voronny's polygons include $\theta N \ln N$ operations (N is the number of points in the domain and θ is some constant) and require a significant memory capacity. Algorithms with number of operations $\theta_1 N^2$, but with small main memory capacity used, can be permitted in realization on minicomputers (for example, the YeS-1010).

Let a field of point objects $\{a_j\}$ be given on a plane. Let us connect the set of all points $x = (x_1, x_2)$ of the plane to each point a_j , for which $R(a_j, x) = \min_{(a_j)} R(a_j, x)$, where $R(a_j, x)$ is the distance from a_j to x . The set

of all points x that satisfy the inequality

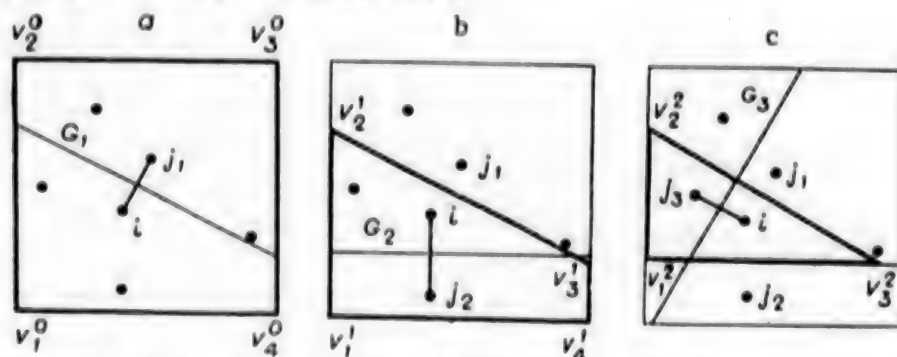


Figure 2. Diagram of Construction of Voronoy's Polygon

of all points x that satisfy the inequality $R(a_i, x) \leq R(a_j, x)$ generates Voronoy's polygons V^1 , which contain point a_i , "closed" from other points a_j ($i \neq j$).

The proposed algorithm is intended for construction of Voronoy's polygons in the limited convex domain of a plane with given coordinates of point objects. For simplicity, let us consider the domain in the form of a square. The basis of the algorithm is the procedure of constructing the polygon for any point of set $\{a_j\}$ without regard to previously constructed polygons. The independence of construction of polygons reduces to $\theta_1 N^2$ operations when constructing the mosaic for N points.

Description of the POLYGON procedure. 0. Let us form the initial polygon V^0 (Figure 2, a) with vertices v_p^0 ($p = \overline{1, 4}$), which coincides with the initial point field.

1. Let us calculate the value

$$r_i^p = \max_p (r^p = \sqrt{(x_i - x_p^v)^2 + (y_i - y_p^v)^2}) (p = \overline{1, m}),$$

where x_p^v, y_p^v are the coordinates of the vertex of polygon V^1 and m is the number of vertices of polygon V^1 .

2. Let us find the point j_{l+1} such that $r_{ij}^2 = (x_i - x_j)^2 + (y_i - y_j)^2$ will be minimal for all values of $j = \overline{1, N}$, except j_q ($q = \overline{1, l}$) for which $r_{ij_1} \leq r_{ij_2} \leq \dots \leq r_{ij_l}$.

3. Let us draw a straight line G_{l+1} passing through the middle of the segment connecting points i, j_{l+1} and perpendicular to this segment (see Figure 2, a for the first iteration of $l = 0$); let us find the points of intersection with sides of polygon V^1 . If G_{l+1} intersects the sides of V^1 , then we go to item 4 and if not we go to item 5.

4. Let us form a new polygon V^{l+1} (see Figure 2, b for $l = 0$ and Figure 2, c for $l = 1$) and let us go to item 1.

5. Let us check the inequality $r_{ij_{l+1}} > 2r_i^p$. If the inequality is fulfilled, then construction of the polygon is stopped and we go to item 6, otherwise we go to item 1.

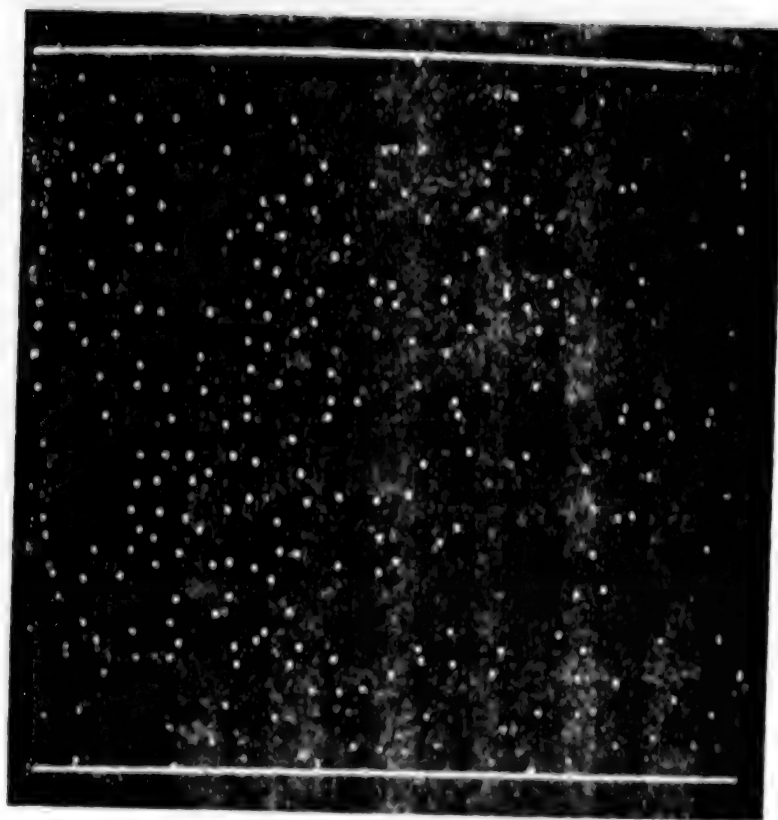


Figure 3. Field of 261 Points Produced by LESART Program on Fragment of Image of Forest Section

6. Let us calculate the area, length of the sides, number of neighbors and the perimeter of a polygon for point i and let us turn to construction of the polygon for the next point.

To accelerate the work of the procedure in fulfillment of item 2, one can do away with calculation of r_{ij}^2 for the points whose coordinates satisfy the inequalities

$$(x_i - x_j)^2 + (y_i - y_j)^2 > r_p^2$$

where r_p is some threshold distance.

Exclusion of items 1 and 5 is also possible if the outcome from the procedure (turning to item 6) is carried out by the threshold number of points KP that do not yield intersections of (x_i, y_i) with γ . Introduction of these heuristics permits one to reduce the number of operations, i.e., the value of θ_1 in $\theta_1 N^2$ and the possible errors essentially do not affect the resulting distributions. For real point fields, the number of errors in construction of one side of the polygon did not exceed 1 percent of the total number of points. The considered algorithm is realized by the VOPLS program. The input data of the program are similar to programs LESRRN and PUASON. The output data are the distributions of the number of adjacent points, the lengths of the sides and the areas of the polygons and perimeters.

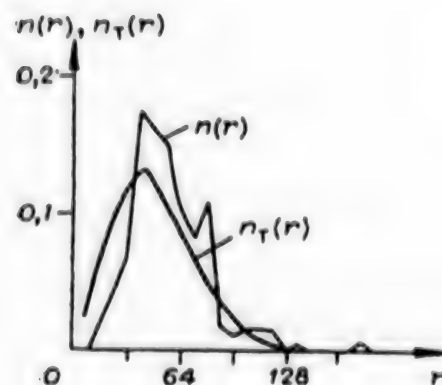


Figure 4. Graphs of Sample and Theoretical Distributions Produced by LESRRN Program: The value of χ^2 for 29 degrees of freedom is equal to 797.86, which contradicts the hypothesis of the random distribution of points. The mean value \bar{r} and the mean square deviation are equal to 48 and 50, respectively.

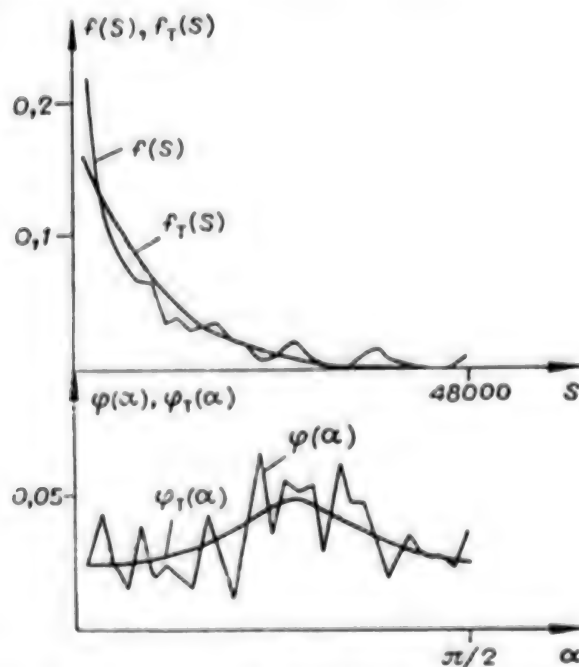


Figure 5. Graphs of Theoretical and Sample Distributions of Areas and Angles $f(S)$, $f_T(S)$ and $\varphi(\alpha)$, $\varphi_T(\alpha)$ found by the PUASON Program: The values of $\chi^2_f = 47.76$ and $\chi^2_\varphi = 31.90$ with $\nu = 29$ permit one to use the zero hypothesis with significance level 0.02.

The first and second moments of the indicated characteristics, the distribution of areas achieved by modelling and also the possible applications of the given program are presented in [3] for the case of a Poisson field. Analysis of Voronoy's polygons for points that are the centers of the crowns of trees

permits one to estimate the status of a forest and the tendencies toward grouping of trees or toward regular distribution with respect to images of forest sections.

The GROUPK program separates the related groups in a set of data that represent a random point field. It is assumed that two points i, j belong to the same group if the distance $r_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}$ between them is less than some threshold distance P_r . Selection of the threshold is very important to find "natural" groups.

Since a fragment of an aerial photograph of a forest measuring $NX \times NY$ is considered as the set of data, then selection of the threshold is related to the number of trees (KD) on this fragment [2]:

$$P_r = K \times NX \times NY / (\pi KD),$$

where K is the given parameter.

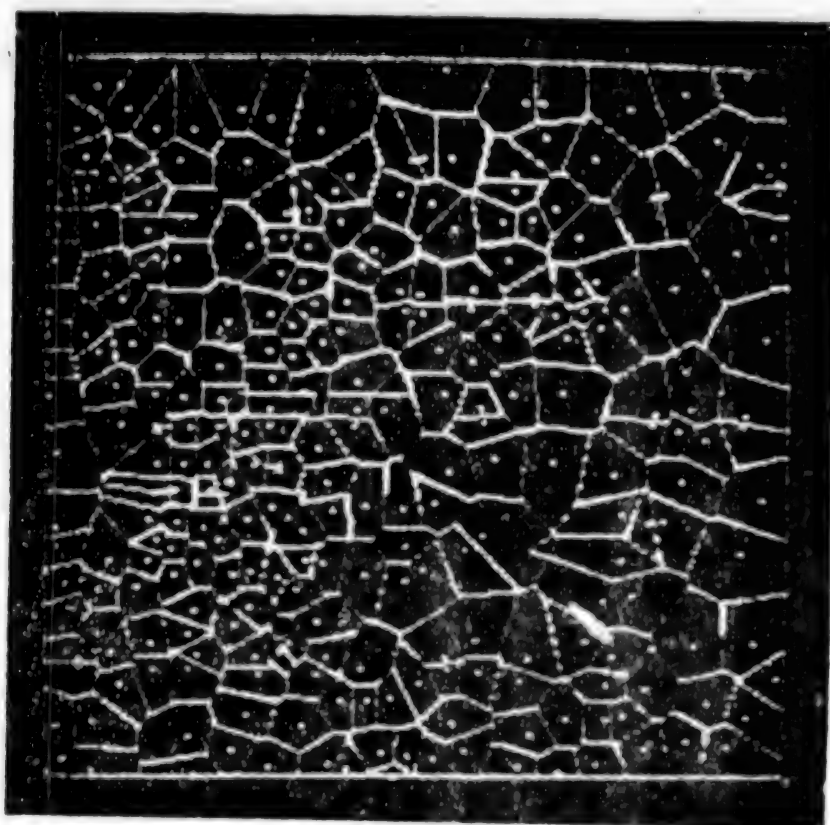


Figure 6. Voronoy's Polygons Constructed by VOPIES Program

The group separation algorithm includes the following. Let us arbitrarily select one of the points as the initial point and let us regard it as the first point of the group. We then find all the nearby points, the distance to which from the first point is less than the threshold distance, and we include them in the group. We then take the second point of the group and find

all the points nearest to it (of those which have not yet been included in this group). We repeat this procedure until the distance to the remaining points will be greater than P_r for each of the points of the group. We then assume some point not included in the group as the initial point for the next group and we repeat the process of formation of groups. The coordinates of the centers (as the centers of gravity of the points falling in the group) are found simultaneously with separation of the groups.

Пороговое расстояние (1)	$P_r = 53$					$P_r = 75$												
Число точек в (2) группе	1	2	3	4	5	1	2	3	4	5	6	8	9	11	13	48		
(3) Число групп	133	44	5	5	1	57	15	8	4	5	2	2	1	1	1	1		

Key:

1. Threshold distance
2. Number of points in group
3. Number of groups

The average number of groups of all dimensions per unit area can be indicated for a Poisson point process [3]:

$$C \approx (\pi P_r^2 \lambda^2) / \exp(\pi P_r^2 \lambda - 1),$$

where λ is the density of the points.

The output parameters of the program for different threshold distances are the following characteristics of the groups: the number of elements, the coordinates of the centers, the mean radius and the mean square deviation of the radius for each group, the average distance between points of the group and the mean square deviation of this distance.

The program permits one to analyze random point fields for group distribution, which is of important significance for analyzing the status and for predicting the development of a forest.

Thus, the software for preliminary statistical analysis of aerial photographs of a forest, also applicable to other images of point objects, has been developed.

The results of the operation of some programs is illustrated by Figures 3-6 and by the table, in which the distributions of groups for different threshold distances P_r are presented. The initial values can be found by the LESRRN program or can be estimated as the mean value of the distance to the nearest neighbor for random distribution.

BIBLIOGRAPHY

1. Nesterikhin, Yu. Ye. et al, "Automated Image Processing Complex," AVTOMETRIYA, No 4, 1980.
2. Kendall, M. and P. Moran, "Geometricheskiye veroyatnosti" [Geometric Probabilities], Moscow, Nauka, 1972.
3. Getis, A. and B. Boots, "Models of Spatial Processes," Leningrad, Cambridge, 1978.
4. Himmelblau, D., "Analiz protsessov statisticheskimi metodami" [Analysis of Processes by Statistical Methods], Moscow, Mir, 1973.
5. Ripley, B. D., "Modelling Spatial Patterns," JOURNAL OF THE ROYAL STATISTICAL SOCIETY, Vol B39, No 2, 1977.

COPYRIGHT: Izdatel'stvo "Nauka", "Avtometriya", 1982

6521

CSO: 1863/2

ORGANIZATION OF SYSTEMS SOFTWARE OF AEROSPACE DATA PROCESSING COMPLEX

Novosibirsk AVTOMETRIYA in Russian No 4, Jul-Aug 82 (manuscript received 4 Feb 82) pp 40-45

[Article by N. S. Yakovenko, Novosibirsk]

[Text] Introduction. The given article is a development of [1]; its purpose is to describe organization of the software of an automated aerospace data processing complex [2], which was realized on the YeS-1010 computer--the control computer of the system.

The software of the complex can be divided into three levels. The baseline lower level program modules (systems software) usually written in Assembler language or in PLR-10 high-level language, are engaged in processing input-output operations or the simplest operations (with minimum calculations) of processing data coming from the scanning device of the Zenit-2 photometric automaton [3]. Medium-level program modules (specialized software), usually written in FORTRAN language and that carry out preliminary processing of incoming data, are formulated in the form of final functional operations that use low-level modules. Applied programs, which usually operate in the dialogue mode or in the dialogue mode with program initiative, which realize a specific algorithm for processing data recorded on photocarriers (specifically, astronomical and aerial photographs), are written or can be developed on the basis of low- and medium-level program modules. The applied programs also include diagnostic monitor systems designed to adjust and debug various control CAMAC units, other functional parts of the photometric automaton and its specialized external devices.

Storage, call and assembly of modules. The text of each program is stored in the symbol file of the FMS-10 file control system [4]; the name of the text module coincides with the name of the first section (if this is an Assembler module or a module written in PLR-10 language) or with the name of the first subroutine (if this is a FORTRAN module). The symbol files are processed by using the text data preparation and editing system [5]. The object module of RB format [6], which will have the name coinciding with the name of the first section or subroutine, and can be placed in one of the libraries of the disk operating system (SL for specialized medium-level modules and UL for low-level systems modules), is produced after translation of the text module, generally containing several sections or subroutines. The name of the object module can

be subsequently changed by using the BIB systems program, but the names of the sections or subroutines cannot be changed. Call of single sections presents no difficulties upon assembly of the executed programs of RMI format by the linkage editor LINKD. If the object module contains several sections and if only the sections different from the first are called from the main program, the linkage editor cannot find them in the library. To avoid these situations, the first section is used in the described modules for loading the entire module and contains either "variable fixing" operations (see below) and variable and equipment initiation operations or simply returns to the call module. It is sufficient to call the initiating section once (at the beginning of the program). If the program to be executed has an overline structure, the call of similar sections can be placed in the trunk of the program (single call) or at the beginning of one of the parallel branches (automatic multiple call with alternate loading of the branch in the main memory).

The program to be executed can operate only in one of two modes: in normal mode (SM--SLAVE MODE) or in the privileged mode (MM--MASTER MODE). Unfortunately, a translator with FORTRAN does not permit the program to be executed to operate in the MM mode if it has a subroutine with parameters written in FORTRAN. All the low-level modules contain input-output instructions and accordingly should operate in the MM-mode. Thus, programs that utilize these modules should operate in the MM-mode and can have no FORTRAN-subroutines with parameters. The parameters can be transferred in the MM-mode to a FORTRAN module only through the common data unit (COMMON) with call without parameters. Call of the Assembler module from the FORTRAN program is possible and is used in the ordinary manner and, if the section simulates the subroutine-function, i.e., it transfers the value in the registers and does not require an independent variable, the actual parameter should then be indicated: A = FUNCT (DUMMY). The method of fixing the variables is used to accelerate the process of parameter transfer. The variables for the input and output data, with which the remaining sections will work, are indicated for this in the call of the initiating section of the module with several sections and these sections are called without parameters (the input data are assigned prior to the call and the result is assigned prior to the return), as when working with a common data unit, except the variables here will be found in the local data segment. The parameter transfer process is accelerated due to the fact that the FORTRAN-translator adds the instruction of loading the X-register to each operation when working with COMMON-variables and the next instruction will be global indirect-indexed instead of a single direct local instruction in the described method.

Organization of systems program modules. All sections of the program to be executed can operate either in the SM- or in the MM-mode as a function of the given parameter for assembly of the RB-modules by the linkage editor. Any systems module services any external device and therefore should operate in the MM-mode. Three different methods are used in the described software. The first standard method of designing the program modules is included in writing the handlers, when the processing program can be logically divided into elementary input-output operations for a specific external device (or, for example, for a UMSO--unified main line data exchange system [7]). The handler

(or rather its second half) operates in the MM mode, while the program that has access to the handler operates both in the MM- and in the SM-mode. The second standard method includes the fact that the systems programs that use privileged instructions were formulated in the form of separate sections and are loaded during generation as supplementary sections of the monitor operating in the MM-mode. The programs can operate in both modes both in the first and second case. The third method includes the following. One can write and place in the monitor two supplementary sections of the monitor during its generation which would replace the SM-mode for the MM-mode and vice versa (by entering the corresponding data in a cell with address 6 of the global data segment of the program to be executed prior to the instruction RSV) so that the program operates in the SM-mode and the systems modules operate in the MM-mode (similar to the first and second parts of the handler). COMMON-variables can be used for parameter transfer. If we want to use the procedure of parameters in the FORTRAN-subroutine, then the program should operate in the SM-mode by this moment (or always) (due to operation of the FORTRAN-translators (FORTD) up to at least the ninth version). Therefore, when using the first two methods, it is no longer necessary for the programs to operate in the MM-mode and accordingly the modules with privileged instructions cannot be used. In the third method, due to the fact that the linkage editor generates addresses with absolute values or with values with respect to the beginning (G) of the program for the entire program as a whole rather than with respect to each module separately, and the supplementary sections that switch the mode also cannot solve problems of parameter transfer to the FORTRAN-module due to the difficulty of writing the Assembler program with the same type of addresses.

The first two methods of operation are acceptable if the systems part of processing as a whole no longer changes, since the process of debugging the handler's and supervisor's sections together with the process of generation is a rather complex procedure. The third method can be applied with special care to methods of addressing for writing the systems programs. Therefore, with the exception of the separately described cases when writing the FORTRAN-programs, assembly is accomplished by the linkage editor in the MM-mode, while either the COMMON-variables or the method of recording the variables described above are used for parameter transfer.

The ordinary and fast interrupt system used in the configuration of the YeS-1010 permits inquiries coming from the external devices of the computer to be processed independently of each other in the order of their priority. The following levels of interrupts and inquiries for processing can be distinguished in the existing complex on the basis of the Zenit-2 photometric automaton: inquiry to read the optical density from the scanning device of the automaton (it is processed by using the fast interrupt system)*;

*

The remaining inquiries, numbered in the order of the decrease of priority, are processed by using the ordinary interrupt system.

the inquiry to end "zone" scanning processing (it comes into the computer after bypass in the interpolation mode of the rectangle with numbering of the optical density at each internal point of the rectangle) is completed;

the inquiry to stop further operation with regard to breakdown of the CAMAC equipment (there is no "X" response signal);

the inquiry for analysis of coordinates (with respect to X and Y) of the movable carriage of the board (it comes in upon passage through the -vicinity of the point of destination);

malfunction of automatic frequency tuning or of the amplitude characteristic for both coordinates;

control interrupt in the UMSO system;

data interrupt in the UMSO system.

The two interrupt levels allocated for working with the UMSO are serviced in the standard manner for the YeS-1010 with the exception that any standard external device is serviced by the same interrupt level. Analysis and processing of the incoming inquiries are formulated in the form of a standard program (the UMSO handler), which is connected to the remaining handlers of the system during generation of the monitor. All the handlers are resident programs, i.e., they are always in the computer memory. Access to the UMSO handler from the program is gained by filling the control unit in the standard manner, where the type of order and the nature of its execution are entered. The program to be executed can operate both in the MM- and in the SM-mode.

Interrupts from the Zenit-2 control CAMAC equipment and of its vicinity (CAMAC-display, color monitor and gray-tone monitor) are serviced by a non-standard method for the YeS-1010 computer. This is caused mainly by the imitation of the main computer memory and by the use of the overline structure of program execution when different interrupt levels and the related baseline program modules can be located in parallel branches. If some functional link does not participate in a given program, then the corresponding service module will not be located in the main memory, freeing it for other purposes. The operation of the baseline program modules that control the corresponding units of the automatic equipment (filling of the contacts of interrupt levels, interrupt deactivation words corresponding to the monitor tables and so on) is initiated by the call of the initiating sections from higher level modules or from the user programs.

If a single baseline module, operating with rapid interrupts, for example, is used in a program having no overline structure, its initiating section is called once at the beginning of the program. If the program structure is overline and for some reasons we do not want to place the initiating section in the trunk of the program, then it must be arranged at the beginning of each branch that uses the remaining sections of the given baseline module. If two (or more) different baseline modules operating with rapid interrupt, for example, are used in the program, then the initiating section must be placed each time when access to sections of the other baseline module is changed.

Readout of optical density. The baseline program module for gathering digital data from the IZENIT photograph, which supports the functioning of the scanning device, permits the following operations to be performed: to set the beam at a given point of the scanning domain and to measure the (mean) optical density in it, to assemble the density values on some fragment of the scanning domain in the buffer of the main memory for further processing, to determine the mean density on the entire fragment or in one direction, to transfer the density histogram from the fragment to the buffer to determine the upper and lower bounds, dispersion and so on.

The CAMAC monitor and half-tone displays. The half-tone data display device based on an analog CAMAC monitor with television screen measuring 256×256 points is available in the Zenit-2 specialized external devices for operational inspection of a section of the scanning field. The monitor can operate only in parallel to the scanning device, which is assigned the zone scanning mode. Low-level program modules that service the CAMAC monitor using a movable carriage (see below) provide dynamic inspection of the photographs and also determination of the coordinates of objects on the photograph.

The displays have self-contained memory [2] that permits storage of data about the television frame, consisting of 384×265 N-digit words. For the gray-tone display $N = 8$ and for the color display $N = 9$. Interpretation of optical density on the gray-tone display is zero for the light section of the photograph and 255 for the dark section. Interpretation is similar to a black-white image: zero for the light section of the corresponding color and 255 for the dark section, in the case of synthesis of a color image for each of the three primary colors (R--red, G--green and B--blue). Technical coloring (i.e., imparting the given optical density of some color shade) by using one of four specially selected color ranges (wedges) is employed to improve the image contrast when using a color monitor. There is a capability during operation of the program in the dialogue mode to change the wedge arbitrarily. The main disadvantage is the absence of the capability of reading the content of the memory--in necessary cases it is supplemented by program: data is also recorded in the working file for subsequent readout while being recorded in the memory.

Direct access to the image. Direct access to the arbitrary section of the image to be processed for a wide range of digital image processing problems, guaranteed by the Zenit-2 automaton, offers significant advantages over sequential readout of the image prior to the beginning of processing (for example, in level-line, formation-boundary tracking and other problems).

The baseline program module for shifting the scanning domain, besides assembly, setting and reading of absolute coordinates, essentially processes one algorithm for transition from a current point (X_0, Y_0) to a given point (X_1, Y_1) . To do this, the difference $(X_1 - X_0, Y_1 - Y_0)$, which the electric motor control modules begin to process, thus moving the carriage to point (X_1, Y_1) , is loaded into the corresponding registers. There are two methods of ending the processing of the shift. There is no single-valued message from the CAMAC module that the "error" $(X_1 - X_0, Y_1 - Y_0)$ has been processed; therefore, after

the carriage is in some vicinity of the point (X1, Y1), a time delay is instituted by program (the first method) and the true location of the carriage is then checked and if the difference is not greater than the permissible difference, the transition is assumed to be completed (otherwise the delay is repeated). The second method includes calculation of the number of character shifts of the increment register to a given value by both coordinates.

Conclusions. The baseline program module system (systems software) is a compatible and easily expandable system that does not duplicate writing of programs similar in the final result; it is accessible to the user programmer who realizes his own algorithm in a high-level language (FORTRAN-4 or PLR-10). The modules are written in Assembler language and are optimized either by the volume of main memory occupied or by speed; they guarantee convenience of operation with external devices in the dialogue mode. The user programmer is freed of the need to become thoroughly familiar with the CAMAC equipment and the control instructions, monitoring and correction of the automaton and its environment, since it can utilize program modules that realize specific input-output functions and preliminary data processing at the call level with parameters, with recording of variables or through common (COMMON) variables. Dialogue and fully automated image processing systems have been and are being developed on the basis of these modules that utilize all the capabilities of the photometric automaton, its environment and the YeS-1010 disk operating system. The capability of using the overline structure of the programs to be executed, which employ the described low- and medium-level modules, permits one to reduce to a known degree the effect of limiting the main memory.

BIBLIOGRAPHY

1. Yakovenko, N. S., "Baseline Software of the Aerospace Data Processing Complex," AVTOMETRIYA, No 4, 1980.
2. Buryy, L. V. et al, "Image Processing," AVTOMETRIYA, No 3, 1980.
3. Nesterikhin, Yu. Ye. and B. M. Pushnoy, "An Automated Image Processing System," AVTOMETRIYA, No 3, 1977.
4. "Sistema obsluzhivaniya faylov na diske FMSp10: Rukovodstvo pol'zovatelya" [FMS-10 Disk File Service System: User Manual], Videoton, 1975.
5. Yakovenko, N. S., "Text Data Preparation and Editing System for YeS-1010 Computer," Novosibirsk, Izd-vo IAIE SO AN SSSR, 1979 (Preprint No 110, Sibirskoye otdeleniye, IAIE, AN SSSR).
6. "Redaktory svyazey: Rukovodstvo pol'zovatelya" [Linkage Editors: User Manual], Videoton, 1974.
7. Bredikhin, S. V. et al, "Input-Output Linkage and Control Software in Unitized Mainline Exchange System (UMSO)," in "Avtomatizatsiya eksperimenta" [Automation of Experiment], Novosibirsk, Izd-vo IAIE SO AN SSSR, 1976.

COPYRIGHT: Izdatel'stvo "Nauka", "Avtometriya", 1982

INTERACTIVE PRINTED-CIRCUIT CARD DESIGN SYSTEM

Novosibirsk AVTOMETRIYA in Russian No 4, Jul-Aug 82 (manuscript received 28 Aug 81) pp 59-63

[Article by V. A. Meleshikhin, D. G. Frizen and Ye. G. Yurashanskiy, Novosibirsk]

[Text] The considered system was developed and is being operated at the IAIE SO AN SSSR [Institute of Automation and Electrometry, Siberian Department, USSR Academy of Sciences] since 1978. The system includes a program pack that provides the following capabilities: input of printed-circuit topology from drawings prepared by the designer, editing of the printed-circuit card topology, automatic configuration, arrangement and routing of printed-circuit cards, issue of designer documentation and control data for machine tools with ChPU [Numerical program control].

Printed-circuit card topology input program. The program permits computer entry of printed-circuit card topology from designer drawings made in arbitrary scale by using some encoder.

During work, the operator has the capability of selecting the grid spacing (with discreteness of 0.025 mm), the tools for conductors and contact surfaces and to determine the set of masks which will be used to draw the given printed-circuit card and the layer in which the components will be located.

There is the capability of determining the groups and macroses to facilitate the work of the operator and to reduce the probability of making mechanical errors.

The group is the filled sequence of actions by the operator to enter or correct the topology of the printed-circuit card. Recursion is possible in determination of a group, i.e., determination of one group may include the call of other groups and macroses.

The macros is a rectangular fragment of the printed-circuit card topology, assigned by indicating the opposite vertices of the fragment on already introduced topology.

Macroses and groups can be retained for subsequent use in other operations.

When the macros or group is called, the coordinates and orientation of the elements of topology are corrected as a function of the point of call and the given angle of rotation of the macros or group.

To correct possible errors, the operator has at his disposal means of editing the entered topology of the printed-circuit card. The operator has the capability of recognizing which components are located at a given point and then of removing, replacing or changing their configuration. Information about the components is sent to an alphanumeric display and includes data on the type of component (conductor or contact surface), its coordinates, the number of the mask by which it is drawn and the layer in which the given component is located.

The correctness of the operator's work (use of legal masks for conductors, the inclination of conductors, the sufficient size of gaps and, if desired, a check of the conformity of the topology to the schematic diagram) is checked during entry; in the latter case, a description of the schematic diagram must be available.

The result of operation of the topology input program is the printed-circuit card topology description file. The file contains data sufficient to generate control programs of machine tools with ChPU. The format of the given file is common for the entire system.

Printed-circuit card topology editing program. The program permits one to edit the existing topology of a printed-circuit card and also to introduce new components. Since the operator is concerned with graphical entities during editing, the use of graphic displays for these purposes is more natural and feasible. Since graphic displays guarantee complete and descriptive drawing of the printed-circuit card topology, the operator's actions in correcting it are simple and natural. Therefore, the probability of making mechanical errors is reduced and the operator's work in correction of the topology is considerably accelerated [1].

There are two main editing modes: directive and screen.

In the directive mode, the operator has the capability of implementing editing actions over the entire card as a whole: of determining macroses and of manipulating them, of controlling the editor operating mode, including selection of the grid spacing (with discreteness of 0.025 mm), of controlling the set of masks for a given card (which corresponds to the photoplotter slide), of establishing the scale and position of the window and the number and order of visualization of the printed-circuit card layers and of shifting and selecting layers. In this mode the editing process is controlled by directives while the necessary actions are determined by the corresponding selection of parameters.

The main work in editing the topology of a printed-circuit card is carried out in the screen mode. In this case a fragment of the card is lighted up on the display screen. The operator has the capability of viewing the entire card by moving the window and by indicating and determining the individual

elements by means of a cursor for subsequent editing actions (removal, replacement and changing of the configuration and addition of components, determination of components and the use of macroses). There is the capability of changing the visualization parameters (number of layers shown on the screen and the order and priority of imaging the layers).

The operator can obtain information on the determined components (the type of component, the number of the mask which depicts it, the layer in which the given component is located and its coordinates).

In the screen mode, the editing process takes place by using a joy stick that controls the position of the cursor and by pressing the functional keys. All editing operations are reflected immediately on the display screen.

Semantic verification can be included if needed; this results in verification of each action of the operator in correction of the topology to conformity of the schematic diagram.

Inclusion of semantic verification increases somewhat the response time of the programs, but considerably increases reliability.

The use of a virtual memory permits one to edit large and compact multilayer cards (measuring up to 800 X 800 mm and the number of components is essentially unlimited).

Introduction of additional sorted fields containing partial data on the components and data on the affiliation of the components to specific clusters considerably reduces the response time of the system, especially with semantic verification included.

The program has a modular structure and permits simple adaptation to different types of displays.

Automatic routing program. The program includes an automatic configuration, partial arrangement of the components and routing of connections for large multilayer printed-circuit cards (measuring up to 800 X 800 mm). The program input data is a description of the schematic diagram, library of modules and structures.

The schematic diagram descriptor file contains data on the coordinates and overall dimensions of the routing field, on the libraries of modules used for a given card and on the structure and also a description of the interconnections in terms of signals.

Each module is described locally; the affiliation of the lead of the module to one of the clusters identified by the name of the signal (for example, "Ground," "Power," "Dump" and so on) is determined. The coordinates of the module, its orientation, the distribution of the clusters by layers and the priorities that determine the sequence of routing the clusters may be included in this description. The modules are identified by their own name.

The modules are arranged and the leads are configured and designated automatically for undefined clusters during routing. Clusters having higher priority are routed first; priority is assigned in the description of the schematic diagram or, if it is not indicated in the description, it is calculated dynamically. Clusters for which the layer in which they should be routed is not indicated in the description are distributed automatically by layers.

Prior to beginning of routing, syntactical and partial semantic verification of the description of the schematic diagram is carried out. The conformity of the topology to the description of the schematic diagram, adherence to the load capability of the modules and the presence of illegal connections and "hanging" signals are verified.

The library of modules contains a description of the configuration of modules of the lead, logic circuit, graphic display on schematic diagram and assembly drawing type, load characteristics and the presence and distribution of standard signals ("Ground" and "Power") by leads. The library of modules can easily be supplemented and corrected.

The structure is the printed-circuit card topology descriptor file and it can also be empty. The structure initially contains the standard framework of the card (the configuration of the plugs and the lead to them, the technological components and the identifying legends) as a rule.

During routing, the program continuously issues data on the connections to be routed at a given moment. The built-in editing devices permit one to control the routing process: to authorize or forbid issue of a routing protocol, to determine the direction of the input flows, to stop routing, to correct the resulting topology and description of the schematic diagram and then to continue routing.

The nucleus of the routing program is the modified Lee algorithm [2, 3], since it usually guarantees the most complete routing of the printed-circuit card compared to other known algorithms (channel, main line, topographic and beam). At the same time it is rather easily modified to take into account some technological restrictions and to increase the quality of the topology (orthogonality for two-layer printed-circuit cards, uniform distribution of conductors, optimization of paths by a number of parameters and so on).

All modifications mainly reduce to selection of the wave propagation law. The nature of distribution of the wave front is determined by being given the delays of front propagation in different directions as a function of certain conditions, being given preferred directions of wave propagation, by the boundaries of the domain of its propagation and "by expulsion" of the wave from some domains.

The Lee algorithm was considerably modified for the phase of plotting the path (reaching the wave target only indicates that the path exists, but does not determine it clearly).

Unlike the methods known to us, the path is plotted in two phases: the path is initially plotted in the stack and the finished path is then laid out from the stack to a discrete working field (DRP). This approach permits one to select the final path from a number of versions and to optimize the path according to certain parameters (the number of bends, the number of junction holes occupied by DRP segments, the density of conductors and so on).

Both the wave propagation law and the principles of selecting the path from a set of variants are determined by the decision table, which permits one to easily change the algorithm, adapting it to different levels or technological restrictions.

The discrete working field, the table of clusters, the ring buffer for wave propagation, the stack for plotting the path and also auxiliary information are located in the virtual memory, which removes essentially all restrictions on the size of the card, its density and the number of components.

The program "cache" above the DRP on the alternative segment is organized to accelerate operation of the program and this makes it possible to reduce most accesses to the DRP to the main memory.

Storage of sufficient auxiliary information about the schematic diagram, remaining from the syntactical and semantic verification phase, permits a diagnosis to be made during routing in terms of the schematic diagram.

During routing, the components are configured by modules on the basis of the information contained in the library of modules about the logic circuit and partial arrangement of equivalent modules at the permutation level.

Unlike the generally accepted opinion about the slow operation of the wave algorithm [4], our results indicate the opposite: with accurate realization, the wave algorithm is rather fast (for example, a card measuring 150 X 180 mm, containing approximately 70 modules, on the order of 200 clusters and more than 1,000 connections is routed in approximately 15-20 minutes of processor time). The completeness and quality of routing are rather good: the percentage of routing comprises no less than 90-95 for two-layer cards if the work is not conducted in the interactive mode.

Output processors. These programs issue designer documentation (schematic diagram, module layout diagram, assembly drawing, drawing of topology and so on) to the graph plotters. The documentation is issued according to the requirements of YeSKD [not further identified].

When control information is issued for machine tools with ChPU, the description of the printed-circuit card topology is converted to the format of machine tools with simultaneous optimization of the trajectory of motion of the actuating members of machine tools. The control information can be received in the form of papertapes or magnetic tapes or can be transmitted to the machine tool control processor through the communications channel.

The system is realized on the NORD-10 minicomputer.

BIBLIOGRAPHY

1. Pearce, N., "Printed Circuit Board Layout Using an Interactive Graphics Display," COMPUTER AIDED DESIGN, Vol 2, No 2, 1970.
2. Lee, C. Y., "An Algorithm for Path Connections and Its Applications," IRE TRANSACTIONS, Vol EC-10, No 3, 1961.
3. Geyer, J. M., "Connection Routing Algorithm for Printed Circuit Boards," IEEE Transactions, Vol CT-18, 1971.
4. Selyutin, V. A., "Mashinoye konstruirovaniye elektronnykh ustroystv" [Computer Design of Electronic Devices], Moscow, Sovetskoye radio, 1977.

COPYRIGHT: Izdatel'stvo "Nauka", "Avtometriya", 1982

6521

CSC: 1863/2

APPLICATIONS

DIALOG WITH COMPUTER CONTINUES

Minsk SOVETSKAYA BELORUSSIYA in Russian 28 Jul 82 p 4

[Interview with Aleksandr Fedorovich Chernyavskiy]

[Excerpts] Several years ago we carried in our newspaper an interview with professor A.F. Chernyavskiy, director of the Scientific Research Institute of Applied Physics Problems at the Belorussian State University imeni V.I. Lenin, on the subject of the development at the Belorussian State University of the "EVOS-BGU" experimental computer teaching system--one of the country's first data-based automated teaching systems.

Considering the social and national economic significance of the problem, the editorial office has again addressed the subject of the computerization of higher education, the more so since in recent years new information has been gained in this field of which a knowledge is useful not only for the specialists but also for the general public.

At this time EVOS, a medal winner at the USSR Exhibition of National Economic Achievements, continues to serve the university, even though its console has already been allocated a place in the university museum. The automated class--the computer hall--organized on the basis of the system as long ago as 1975 and which has now performed for more than 6,000 hours, is still successfully functioning.

In 1977 a new teaching system was developed, but this one was not as unique as its predecessor and was designed to be copied, coupled to a computer and training television receiver, and oriented on multifunctional use in educational establishments. The new system is called ATOS-BGU (automated television teaching system of the Belorussian State University). An automated class--the computer hall--has been organized on the base of ATOS. Another three of these systems have been built and are now in the startup and debugging stages. As already mentioned, the USSR Ministry of Higher and Secondary Specialized Education has planned for this system to go into series production in 1984 to supply the leading VUZ's.

[Question] Aleksandr Fedorovich, could you describe the new automated television teaching system in more detail?

[Answer] Of course. ATOS is made up of 15 display consoles for the students and a teacher's console. Each console is a display unit consisting of a keyboard and a Yunost'-402 television set. The group devices are coupled by a centralized control display for the class and by the hookups with the computer. A person who knows the series-produced display units will be struck by the compactness of the ATOS displays, the expanded set of keyboard keys, and the use of television receivers for display.

The essence of the compact display in the ATOS system is hidden behind these fine external appearances. The compactness is achieved by centralizing a substantial part of the equipment into group devices, which has reduced the cost of each working station by a factor of 2 to 3 compared with the series-produced displays. The expanded set of keyboard keys indicates additional functions for the displays that are so essential for studies in different disciplines. ATOS has the following functions: recording indexes for mathematical expressions and chemical formulas, producing graphics information, data contrast separation (negative and positive), fast setting of multiple symbols expressions and a number of other facilities. Use of the television receivers reflects the most important feature of ATOS--integration of the automated teaching system and training television.

Recognition of the new system has been convincingly seen in medals of the USSR Exhibition of National Economic Achievements, diplomas of the Belorussian SRR Exhibition of National Economic Achievements, and participation in national exhibitions of the USSR and Great Britain (1978) and international fairs in Leipzig (1981) and Plovdiv (1981).

Taking into account that the requirement for a major VUZ is about 100 display units, the expected savings from the introduction of each VUZ ATOS system will be hundreds of thousands of rubles.

9642

CSO: 1863/350

METHOD OF ORGANIZING INTERACTIVE MODE FOR CONDUCTING AUTOMATED LABORATORY GENERAL PHYSICS EXPERIMENT

Novosibirsk AVTOMETRIYA in Russian No 4, Jul-Aug 82 (manuscript received 27 Aug 81) pp 7-13

[Article by A. N. Vystavkin, Yu. V. Obukhov and V. V. Romanovtsev, Moscow]

[Text] Introduction. Development and investigation of interactive systems is one of the most important directions of development of work in the field of automation of scientific research. Definite success has been achieved in development of interactive control [1], design [2], data processing [3] and automated programming [4] systems. Systems for automation of most general physics experiments, however, have a number of differences. We note two of the most important here in our view. The first includes the need to rather frequently change the layout of the experiment and the functional structure of the installation, which results in changing the automation hardware-software; the second is the need to automate as large a number of diverse experiments conducted in a laboratory or institute on the basis of the same hardware-software. Problems of development of standardized hardware-software to sufficient degree for conducting experiments in the interactive mode become primary in this regard in laboratory systems.

The structure and composition of hardware and software that realizes the interactive mode of operation of a laboratory system for automation of general physics experiments are described in this paper. We determined this structure in the first phase, basing it on the experience of developing a laboratory system for automation of diverse experiments in the field of physical electronics contained in the collective-use measuring computer system (IVSKP) of the Institute of Radio Engineering and Electronics, USSR Academy of Sciences. The IVSKP is based on a network of Nord-10/S minicomputers and the CAMAC sequential main line.

Structure of the hardware-software. The functional composition of laboratory systems for automation of experiments has been rather widely illuminated in the literature (see, for example, [5, 6]). Laboratory systems usually include equipment control subsystems of the experimental installation and data gathering, processing, display and storage subsystems. The most important from the viewpoint of organizing the interactive mode of conducting experiments are data control, gathering and display subsystems. Our task was to develop the

hardware-software of these subsystems which would be independent to a significant degree of the type of experiment and which would permit us to realize the interactive mode. We feel that the experimenter should have the capability of selecting different devices (from those available in the system) in this mode to control the experiment and to measure physical values, to assign the operating modes of these devices at his own discretion, to select different devices for graphical display of data both during the measurements and after them, to assign the function and type of graph to display it and to conduct the experiment, i.e., to decide which operations should be performed (for example, to calibrate the devices, to make measurements, to process data and display it on the selected device, to enter data into the archives and so on).

In this part, let us first describe the selected configuration and composition of the hardware of the laboratory automation system and then let us select the structure of the developed software of the control, gathering and display subsystems and let us finally present a typical block diagram for the user program, based on the described systems support and that realizes the interactive operating mode.

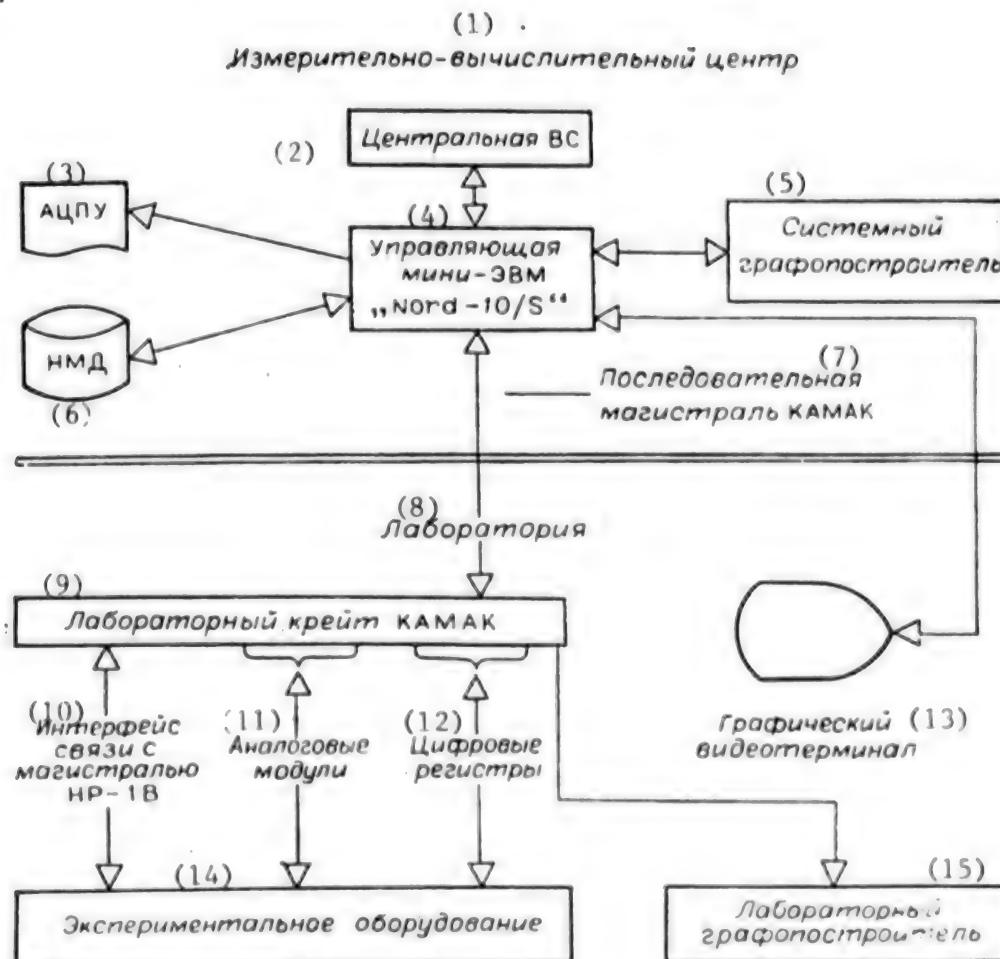


Figure 1.

[Key on following page]

[Key continued from preceding page]:

- | | |
|-----------------------------------|--|
| 1. Measuring computer center | 9. CAMAC laboratory crate |
| 2. Central computer network | 10. Interface for communicating with HP-1B main line |
| 3. Alphanumeric printer | 11. Analog modules |
| 4. Nord-10/S control minicomputer | 12. Digital registers |
| 5. Systems graph plotter | 13. Graph video terminal |
| 6. Disk unit | 14. Experimental equipment |
| 7. CAMAC sequential main line | 15. Laboratory graph plotter |
| 8. Laboratory | |

Hardware. A block diagram of the hardware for the automation system is presented in Figure 1. The Nord-10/S control minicomputer communicates with the experimental installation by using the CAMAC sequential main line. The CAMAC communications interface was selected for several reasons, the main one of which is the capability of using a wide range of serial modules. The baseline set of devices and modules includes a 40-channel relay multiplexer for switching of analog signals, a systems multimeter, an analog-digital converter with regulated amplification factor, a multichannel digital-analog converter, modules for communicating with interfaces in other standards--HP-1B and TTU, output and input registers for control of relays, digital circuits and devices with nonstandard interfaces, a graph plotter and a graph video terminal.

The user communicates with a computer by using a Tectronix-4025 graph video terminal. The region of the terminal screen can be dynamically divided into a working zone, which is used for real-time graphical display of data, and a monitor zone for independent alphanumeric dialogue with the computer. Operations can be conducted on any computer included in the IVSKP network, specifically a computer of more powerful configuration designed to make theoretical calculations and for modelling, by using the system for intercomputer communications with this terminal. Moreover, one can retrieve data to IVSKP peripheral devices (for example, one can plot high-quality graphs on the systems graph plotter). Figuratively speaking, the laboratory terminal is the "window" to the IVSKP.

Software. The software for interactive conducting of an experiment, the block diagram of which is shown in Figure 2, is based on two subsystems--a real-time data display subsystem and a hierarchical library of control subroutines for CAMAC devices and modules.

Data display subsystem. The standard display software consists of graph display instructions and the library of Plot-10 graph subroutines. Graph pack subroutines are usually inserted into the user's applied program for graphical data display and some packs [7] include user-oriented graph formulation subroutines (plotting and numbering of axes, imprinting legends and so on). The main disadvantage of this approach is the need to modify the user program each time when graph formulation must be changed--the type of axes, the content and location of legends and so on. We organized the graph subsystem so that the user assigns all the service information about the graph (the graph parameters file) before operation of the applied program using the dialogue graph plotting and

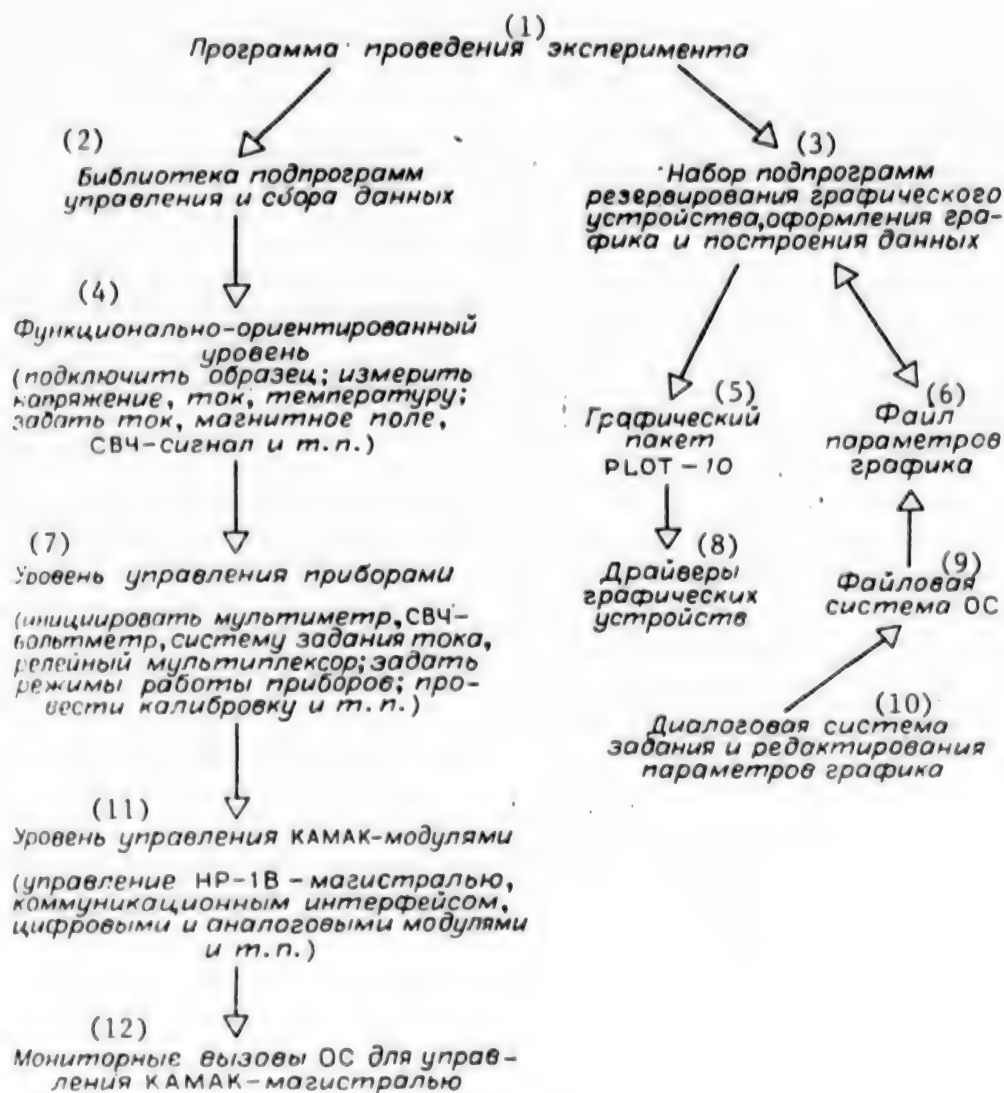


Figure 2.

Key:

1. Experiment running program
2. Library of data control and gathering subroutines
3. Set of reserve programs of graph device, for graph formulation and for plotting of data
4. Functional-oriented level (connect the model, measure voltage, current and temperature, assign current, magnetic field, SHF signal and so on)
5. Plot-10 graph pack
6. Graph parameter file
7. Device control level (initiate multimeter, SHF voltmeter, current assignment system and relay multiplexer, assign device operating modes, carry out calibration and so on)
8. Graph device drivers

[Key continued on following page]

[Key continued from preceding page]:

9. File system of operating system
10. Dialogue system for assignment and editing of graph parameters
11. CAMAC module control level (control of HP-1B main line, communications interface, digital and analog modules and so on)
12. Monitor calls of operating system for control of CAMAC main line

editing system. This file is not dependent on the type of graph device and is stored on magnetic disks. The special graph formulation subroutine, which contains the name of the graph parameters file and the number of the graph device as formal variables, should only be called from the applied program. It is natural that there is access to the parameters file from the applied program. This structure of the display subsystem permits easy changing of the type and form of graphs, selection of the necessary graph device and also frees the user program of drudge work in graph formulation and frees the user himself of the need to study tens of graph subroutines. Moreover, the graph form of interaction of the investigator and experiment is realized by linking the graph parameters to the physical parameters of the experiment and the method of organization of this interaction is standardized to a significant degree, i.e., it is not dependent on the type of graph and experiment.

Device control subsystem. The Sintran-III/VS disk operating system of the Nord 10/S control minicomputer has several monitor calls to control the CAMAC main line, which can be used to execute CAMAC transmissions, to start programs by LAM signals, to analyze the state of the crate controller status register and so on. However, to increase the flexibility of the user control programs, they can be constructed on the basis of a multilevel subroutine pack that realizes different experimental functions. This approach permits one to develop a basic library of hardware-software modules for experiment automation at different laboratories of the institute.

We developed a set of subroutines for devices and CAMAC modules that are contained in the described system (see Figure 2). Functional-oriented subroutines that realize different typical experimenter actions (switching on the measuring current, measuring voltage and so on) are included at the upper level of the hierarchy. These actions are realized by one or several physical instruments and devices which are usually controlled by several CAMAC modules. The functional-oriented subroutines are based on instrument-oriented subroutines for control of the systems multimeter and SHF voltmeter, measuring current and magnetic field assignment systems, for control of SHF devices and other functional systems of the experiment. These program modules utilize CAMAC module and HP-1B main line control subroutines. Monitor calls for control of the CAMAC main line of the operating system are used at the lowest level of the hierarchy. The pack is written in FORTRAN-77, which permits easy changing of the subroutine if necessary.

Typical block diagram of program for interactive conducting of experiments. The typical block diagram of the experiment-servicing program is presented in Figure 3. A menu of subsystems for introduction of some experimental and

technological parameters, initialization of the system, making measurements and analysis of data is presented to the user at the beginning of operation of the program.

The data input subsystem. This subsystem is intended first to enter service information about the object to be investigated (technological parameters for preparation of specimens, number of the specimen, data of conducting the experiment and so on) into the archive and second for input of some values of experiment parameters (the maximum permissible values of parameters for the given specimens, the number of specimens and their conformity to relay multiplexer channels and so on). The initial values are assigned "by omission" of experimental parameters.

The subsystem for initialization of instruments and for making calibration measurements. This subsystem is designed to check the readiness of the instruments and CAMAC modules, to calibrate the instruments and special control devices and to make calibration measurements to determine the normalizing coefficients.

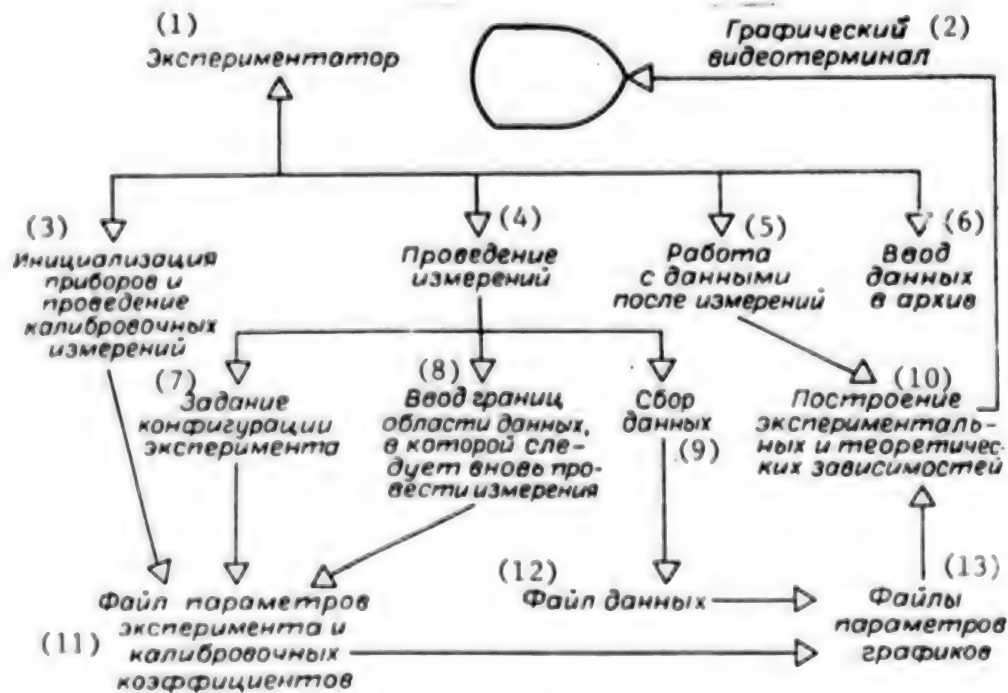


Figure 3.

Key:

1. Experimenter
2. Graph video terminal
3. Initialization of instruments and making calibration measurements
4. Making measurements
5. Working with data after measurements
6. Entry of data into archive
7. Assignment of experiment configuration

[Key continued on following page]

[Key continued from preceding page]:

8. Input of boundaries of data region in which measurements should again be made
9. Data gathering
10. Plotting of experimental and experimental functions
11. Experiment parameters and calibration coefficients file
12. Data file
13. Graph parameters files

When working with this subsystem, the experimenter adjusts the equipment, selects the optimum parameters and signals in the manual mode (for example, he selects in manual mode the appropriate amplitude of the probe signal with analog measurement of derivatives of volt-ampere characteristics and makes calibration measurements in the automatic mode on an ohmic resistor to determine the normalizing coefficients).

Subsystem for making measurements. This subsystem permits the experimenter to assign the experiment configuration in the control mode by using directives, i.e., to select the measurement devices (for example, voltage can be measured with a systems multimeter or an analog-digital converter, which has greater measurement speed but worse resolution); it permits him to select devices for real-time data display (graph plotter, graph video terminal or recording oscillograph controlled by the CAMAC module), to issue instructions for the beginning of measurements or to interrupt measurements, to assign a new region of physical variables in which measurements should again be made with plotting of data on a newly formulated graph after making measurements and display of data in graph form (we call this the graph form of investigator-automated experiment interaction).

As an example, let us describe one of the phases of the experimenter's work when investigating weakly-bound superconductors. Copies of the screen images manufactured on a graph plotter, of the graph video terminal, produced during measurements of the volt-ampere characteristics of Josephson junctions, are presented in Figure 4. After measuring one of the curves (the dashed curve), the experimenter issues instructions to switch on the SHF field and to repeat the measurements. The steps of current appeared on the volt-ampere characteristics of the irradiated junction (the solid curve). The experimenter can issue the instruction WINDOW and can assign the boundaries of the "window" of the parameters, after which the characteristic is measured in the new range of parameters (the curve on the inset).

Subsystem for working with data after measurements. In this subsystem the user can plot the graphs of different cross-sections of the range of experimental data and can compare them to theoretical functions. In this case, by assigning different combinations of possible mechanisms and by plotting the theoretical functions on the same graph with experimental data, he can compare the contribution of these mechanisms in the different zones of experimental parameters.

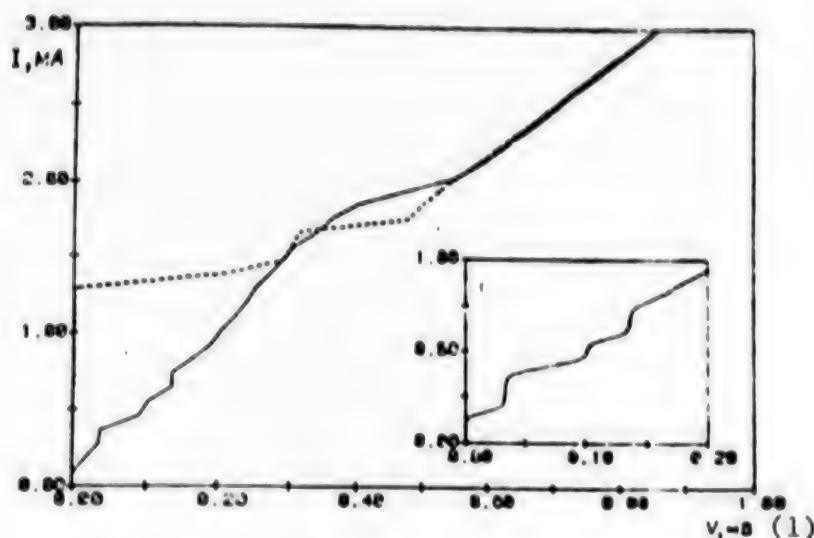


Figure 4.

Key:

1. Millivolts

Let us present an example of working with data of investigations of the superconductor-normal metal point contact [8]. The experimental dependence of the normalized differential conductance at zero voltage on the normalized temperature is shown by the points in Figure 5. The top solid curve corresponds to the theoretical behavior of conductance of the short-circuited contact while the bottom curve corresponds to contact with the insulator layer. The conductance of a real-contact can be determined by some combination of these mechanisms. Being given different values of their ratio and by comparing the plotted function to experimental data (the dashed curves), the experimenter ascertains the absence of other mechanisms and determines the quality of the specimen. He can then issue instructions to plot the selected function on the graph plotter for an article or report.

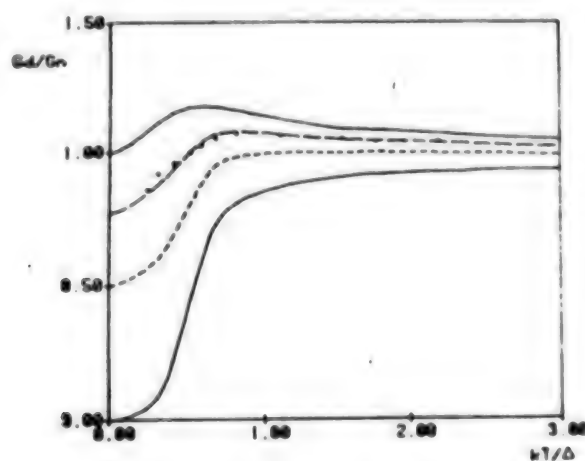


Figure 5.

After completion of work in the subsystem, the experimenter can indicate which data should be entered in the archives.

Conclusions. Based on the described systems hardware-software and on the principles of organizing experiment conducting programs, systems were developed for automation of various types of experiments in the field of SHF cryoelectronics that permit one to make measurements of the volt-ampere characteristics in the interactive mode of many (up to 40) specimens at different temperatures, magnetic fields and levels of SHF output.

As an example let us indicate experiments to determine the properties of superconducting arm tunnel junctions and to investigate superconductor-normal metal point contacts and noise in Josephson point contacts. The volt-ampere characteristics of the junctions (and there may be more than 10 on the same substrate in the case of the arm version) must be measured in different ranges of current and voltages, in different magnetic fields, at different temperatures and different levels of SHF effects to determine the characteristics of the specimens. If one takes into account that the time constant of measurements at the same point can be large (on the order of tenths of a second) due to the low signal level, then conducting all the measurements requires much more time than that during which liquid helium boils in a Dewar flask. In the interactive mode, the experimenter, having determined roughly the main characteristics of the specimens by measuring them in the scanning mode with sufficient resolution, can select the most interesting specimens and ranges of parameters and can make detailed measurements for them.

We note in conclusion that the described principles and devices are also used in other laboratories of the institute serviced by the IVSKP, since the same graphical display hardware-software and many of the described CAMAC modules are used in them (and the system can be expanded if other CAMAC modules are used).

BIBLIOGRAPHY

1. Sheridan, T. B. and W. R. Ferrell, "Sistemy chelovek-mashina" [Man-Machine Systems], Moscow, Mashinostroyeniye, 1980.
2. Person, B., "An Interactive Framework for Design Automation," AUTOMATIC PROGRAMMING, Vol 9, No 2-3, 1980.
3. Ling, R. F., "General Consideration on the Design of an Interactive System for Data Analysis," COMMUNICATIONS OF ACM, Vol 23, No 3, 1980.
4. Negus, B., M. J. Hunt and J. A. Prentice, "DIALOG: A Scheme for the Quick and Effective Production of an Interactive Application Software," SOFTWARE PRACTICE AND EXPERIENCE, Vol 11, 1981.
5. Cole, H., "System/7 in Hierarchical Laboratory Automation System," IBM SYSTEMS JOURNAL, No 4, 1974.

6. Vystavkin, A. N., "Computer Equipment in Physical Investigations," VESTNIK AN SSSR, No 4, 1979.
7. Butland, J. and S. D. Butland, "An Easy-to-Use Graph Drawing Package," COMPUTER, Vol 13, No 2, 1980.
8. Gubankov, V. N. and N. M. Margolin, "Nonlinear Electrical and Superhigh Frequency Properties of Superconductor-Normal Metal Microcontacts," ZHETF, Vol 80, No 4, 1981.

COPYRIGHT: Izdatel'stvo "Nauka", "Avtometriya", 1982

6521

CSO: 1863/2

MULTITERMINAL SYSTEMS FOR ACADEMIC RESEARCH WORK AT TECHNICAL VUZES

Novosibirsk AVTOMETRIYA in Russian No 4, Jul-Aug 82 (manuscript received 13 Jan 82) pp 13-17

[Article by A. I. Druzhinin, A. G. Kozachok, A. V. Loginov and V. N. Sarnadskiy, Novosibirsk]

[Excerpts] Multiterminal systems based on the Elektronika-60 microcomputer and the SM-4 minicomputer, developed at the Novosibirsk Electrical Engineering Institute and oriented toward use in the training process in a technical vuz, are considered in the article.

A block diagram of the system based on a microcomputer is presented in Figure 1. Besides the Elektronika-60 processor, it includes a main memory (OZU) of 28 K 16-digit words, an SM-5603 external floppy disk memory (UVPGMD), eight VTA-2000 video terminals, a DZM-180 sequential alphanumeric printer (ATsPU) (replacement by the similar DARO-1156 device is possible), an FS-1501 papertape input device and a timer required for operating in the time-sharing mode.

The interface sequential printing microcomputer integration units (BSPP) and the floppy disk unit (BSNGMD) are based on the design of TEZ cards of the Elektronika-60 microcomputer and are printed-circuit cards measuring 252 X 148 X 12 mm located directly in the common bus (OSh) unit. The group integration unit (BSG) that permits connection of seven alphanumeric video terminals to the computer channel is made in the form of a structurally self-contained unit which is connected to the common bus by an adapter cable. It is a chassis with RPPM16-278 plugs and five printed-circuit cards inserted in it: the control device card connected by the adapter cable to the Elektronika-60 microcomputer and four cards for communications of the assemblies with the video terminal (two communications assemblies each are located on each card).

The Elektronika-60, the SM-5603 floppy disk external memory device, the FS-1501 photoelectric papertape input device, the VS-5-10A and the VS-27-6 power supply units rated at 5 and 27 V, the BSPP, BSNGMD and BSG interface units and the main memory unit rated at 32K are located in a standard ASVT-M [modular computer equipment system, version M] bay. An external view of the system is shown in Figure 2.

The operating system that supports the multiterminal operation of the computer system is the RAFOS real-time disk operating system [3]. The multiprogram dialogue system based on BASIC language operates under control of the RAFOS.

The BASIC used in this system has some characteristics absent in the standard version of BASIC. These characteristics include:

- apparatus for working with files, which permit organization of program and data files in external memories;

- virtual data files;

- the presence of integer and text constants and variables, besides real integers;

- segment and overline organization of programs;

- the capability of calculations both with simple and with double accuracy.

The simplicity of BASIC language and also the dialogue mode of operation of the multiterminal system makes it very convenient for teaching programming to students, for conducting exercises in a number of disciplines and for performing course and diploma work.

The developed system passed experimental operation for a semester and was turned over to the chair of information measuring equipment.

The system based on the SM-4 minicomputer is designed for use mainly in senior courses. A block diagram of the system (the minimum complex) is presented in Figure 3. It includes a central processor (TsP), a main memory of 128 K words, an IZOT-1370 disk unit (NMD), a mosaic printer (UMP), from 8 to 16 VTA-2000 video terminals (VT) and an interface expander (RIF) required to connect additional terminals.

The RSX-11M/V3.1 operating system is used to support multiterminal operation. Its main function is to guarantee control of the shared resources of the system and to distribute these resources among problems prepared by users.

COPYRIGHT: Izdatel'stvo "Nauka", "Avtometriya", 1982

6521

CSO: 1863/2

PUBLICATIONS

[Text] Table of Contents of Journal 'AVTOMETRIYA', July-August 1982
Novosibirsk AVTOMETRIYA in Russian No 4, Jul-Aug 82 pp 1-2

CONTENTS	Page
Automation Hardware and Software	
V. A. Vittikh and O. P. Skobelev, Preliminary Data Processing and Gathering Systems in CAMAC Standard for Automated Scientific Research Systems for Physical Fields	3
A. N. Vystavkin, Yu. V. Obukhov and V. V. Romanovtsev, Method of Organizing Interactive Mode of Conducting Automated Laboratory General Physics Experiment	7
A. I. Druzhinin, A. G. Kozachok, A. V. Loginov and V. N. Sarnadskiy, Multiterminal Systems for Academic Research Work in Technical Vuz	13
O. V. Vyushin and P. L. Khrapkin, Standard Subroutine Pack for Working with CAMAC Equipment	17
M. A. Kurilov, V. V. Manako, A. I. Nikitin and I. V. Chichkan', Standard Graph Pack GRAS. Display, Storage and Graphical Data Transmission Devices	23
V. N. Verkhoturov and V. I. Zabolotskikh, Automated Problem-Oriented Complex for EPR Spectroscopy	28
Image Processing	
V. A. Ivanov and G. A. Ivanchenko, Software for Statistical Analysis of Aerial Photographs of Timber	34
N. S. Yakovenko, Organization of Systems Software for Aerospace Data Processing Complex	40
A. G. Kozachok, Problems of Automation of Strength Investigations Based on Coherent Optics and Holography Methods	45
Design Automation	
V. G. Lebed', A. N. Lyubimov and V. V. Rusanov, Method of Display and Visualization of Shape in Dialogue Computer Design Systems	52

V. A. Meleshikhin, D. G. Frizen and Ye. G. Yurashanskiy, Interactive Printed-Circuit Card Design System	59
S. V. Gorin, DISAP--Dialogue Graphical System for Electronic Equipment Design Automated Systems	63
Applied Program Packs	
V. P. Il'in and V. A. Kateshov, EFIR Program Pack for Calculation of Potentials and Their Perturbations	67
A. V. Gavrilin and V. P. Il'in, EDIP Program Pack for Automation of Solving Problems of Electrodynamics	74
V. Ya. Ivanov, Generation of Program Packs with Dynamic Structure Based on Macroprocessor	80
V. T. Astrelin and V. Ya. Ivanov, Numerical Modelling of Collision Processes in Relativistic Electron Gun Accelerators	87
Brief Communications	
M. N. Bukharov, V. M. Vukolikov, A. N. Vystavkin, A. D. Morenkov, A. Ya. Oleynikov, M. I. Pertsovskiy, A. A. Snegirev and N. A. Tikhomirov, Use of Autonomous Controller of CAMAC Crate Based on INTEL-8080 Microprocessor in Experiment Automation Systems	94
V. A. Astaf'yev, A. A. Vinogradov, V. A. Voronov, E. I. Kebin, V. V. Paramonov and V. G. Sukharevskiy, Measurements Automation System on Cyclotron of Scientific Research Institute of Nuclear Physics, Moscow State University	98
M. P. Grishin, Sh. M. Kurbanov and Ye. I. Chernov, System for Automatic Photometry of Half-tone Images Based on Avalanche Photodiode with Operational Amplifier	99
V. I. Guzhov, A. I. Druzhinin, A. G. Kozachok and A. V. Loginov, Measuring Computer System for Investigation of Stress-Strain State of Objects	102
V. P. Kosykh, A. I. Pustovskikh and N. S. Yakovenko, Program Realization of Morphological Processor	103
A. G. Yermolayev, Optimization of Morphological Algorithms for Image Identification	105
Yu. M. Krendel' and Z. B. Kurglyak, A Method of Calculating a Two-cycle Buffering Mode	107

COPYRIGHT: Izdatel'stvo "Nauka", "Avtometriya", 1982

6521

CSO: 1863/2

- END -

- 60 -

END OF

FICHE

DATE FILMED

8 DEC 82